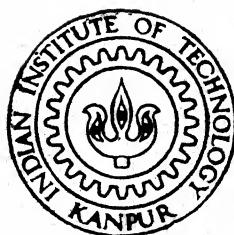


NAVIER-STOKES SOLVER USING HIGHER ORDER INTERPOLATION FOR COMPRESSIBLE FLOW PAST CASCADES

by
GIRISH H. BHANDARI



AE Th
1998 AE / 1998 / M
1998 B 469 n
M
BHA
NAV

DEPARTMENT OF AEROSPACE ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY KANPUR
APRIL, 1998

NAVIER-STOKES SOLVER USING HIGHER ORDER
INTERPOLATION FOR COMPRESSIBLE FLOW PAST
CASCADES

*A Thesis Submitted
in Partial Fulfillment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY*

by
GIRISH H. BHĀNDARI

to the
DEPARTMENT OF AEROSPACE ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, KANPUR-16.

April 1998

21 MAY '98 / AF

CENTRAL LIBRARY
IIT KANPUR

For No A 125505

AE-1998-M-BHA-NAV

Entered in System

Nimisha
25.6.98



A125505



CERTIFICATE

It is certified that the thesis work entitled *Navier-Stokes Solver Using Higher Order Interpolation for Compressible Flow past cascades* by Girish H. Bhāndari has been carried out under our supervision, and that it has not been submitted elsewhere for a degree.

T. K. Sengupta
(Dr.T.K.Sengupta)

CFD Lab

Department of Aerospace Engineering

Indian Institute of Technology

Kanpur 208 016

India

April 1998

R. K. Sullerey
(Dr.R.K.Sullerey)

Propulsion Lab

Department of Aerospace Engineering

Indian Institute of Technology

Kanpur 208 016

India

April 1998

Acknowledgement

I would like to take this opportunity to acknowledge all those who encouraged me and helped me, knowingly or unknowingly, in making this thesis endeavour successful.

First and foremost, I extend my deepest thanks to my parents who are the constant source of encouragement throughout my education.

My utmost gratitude goes to my advisor, Dr.T.K. Sengupta, who has been extremely patient, helpful and caring throughout my work. The success would not have been possible without his encouragement, guidance and insight. I extend my special thanks to him for sharing his valuable ideas and experience with me. I would also like to thank Dr.R.K.Sullerey for his time to time guidance and advise. I am thankful to all professors who taught me during my academic curriculum at *IITK*.

This thesis would remain incomplete if I miss the opportunity to extend my wholehearted thanks to Manoj Nair for his unaccountable help and support. To. Sukanta Majumdar for providing me a basic grid generation code.

I am also greatly indebted to my friends and colleagues from *Maharashtra Mandal*, Hall-4 and *AERO – CFD* lab, specially, Prasad, Rajendra, Adish, Amol for their constant help and support which made my stay at *IITK* memorable.

Finally, my sincere thanks to Indian Institute of Technology, Kanpur, for providing excellent environment for my studies and to *ARDB* for supporting my thesis work throughout my graduation.

Nomenclature

q	:	Unknown Flux
u	:	Tangential Velocity
v	:	Normal Velocity
E	:	Total Energy
p	:	Pressure
H	:	Enthalpy
T	:	Tempareture
Pr	:	Prandtl Number
Re	:	Reynolds Number
S	:	Cell Area
Δt	:	Time step
C_p	:	Specific Heat Constant at constant Pressure
C_v	:	Specific Heat Constant at constant volume
c	:	Local Speed of sound
M	:	Mach number
R, S	:	Eigenvectors
A, B	:	Jacobian flux matrices

Greek Symbols:

γ	:	Specific Heat ratio
----------	---	---------------------

ρ	: Density
ξ	: Abscissa in tranformed plane
η	: Ordinate in tranformed plane
μ	: Viscosity
τ	: Shear stress
σ	: Energy loss
κ	: Thermal Conductivity
ϵ	: Order defining constant for an equation
Γ	: Domain boundary
Λ	: Eigenvalue matrix
λ	: Split Flux Matrix in ξ direction
Δ	: Split Flux Matrix in η direction
α	: Inflow angle
β	: Outflow angle

Subscripts:

∞	: Free-stream quantities
i,j	: Cell Coordinates
c	: Convection terms
v	: viscous terms
o	: Reference quantities

Superscripts:

$*$: Dimensional quantities
$+$: Positive value
$-$: Negative value
\pm	: Positive-Negative combind values
\rightarrow	: Cartesian plane quantities

- $\hat{\cdot}$: Transformed plane quantities
- L : Left of the surface considered for a given cell
- R : Right of the surface considered for a given cell
- T : Top of the surface considered for a given cell
- B : Bottom of the surface considered for a given cell

Contents

List of Tables	x
List of Figures	xi
Abstract	1
1 Introduction	2
1.1 Role of CFD in Turbomachinery	2
1.2 Aim of the Thesis	3
1.3 Thesis Overview	4
2 Governing Equations for Compressible Flows	7
2.1 Introduction	7
2.2 Finite Volume Formulation for Compressible Flow Equations	9
2.2.1 Introduction	9
2.2.2 Nondimensionalization of the Quantities in Flow Equations . .	12
2.2.3 Nondimensionalizing the Inflow Boundary Conditions	14
2.3 Flux Vector Splitting Scheme	15
2.3.1 Introduction	15
2.3.2 Multi-Dimensional Flux Vector Splitting	15
2.3.3 Steger and Warming Flux Vector Splitting	21
2.3.4 Properties of Split Flux Vectors	23

2.4	Interpolation Scheme	25
2.4.1	Introduction	25
2.4.2	High-Order Upwind Scheme- MUSCL approach	25
2.4.3	Free Stream Preservation for the Flux Calculations	28
2.5	Time Integration Scheme	30
2.5.1	Introduction	30
2.5.2	RRK Formulation	31
2.6	Boundary Conditions	33
2.6.1	Periodic boundary condition	33
2.6.2	Solid wall boundary condition	34
2.6.3	Far-field boundary conditions	39
2.7	Spectral Analysis of FVS Scheme	40
2.7.1	Introduction	40
2.7.2	Spectral Analysis	41
3	Grid Generation	47
3.1	Introduction	47
3.2	Algebraic Grid generation	48
3.2.1	Introduction	48
3.2.2	One-Dimensional Stretching Functions	49
3.3	Elliptic Grid generation	51
3.3.1	Introduction	51
3.3.2	Co-ordinate System Generation	51
3.4	Results	55
4	Results and Discussion	62
4.1	Cases of Studies	63

4.1.1	Euler Solution for NACA0012 Cascade domain	64
4.1.2	Navier-Stokes Solution for NACA0012 cascade Domain	65
4.1.3	Navier-Stokes Solution for DCA cascade domain	65
4.2	Conclusions	67
4.3	Future Scope of the Work	68
4.4	Figures	69
	Reference	86

List of Tables

2.1	Phase error for different Interpolation schemes	44
3.1	Values of the constants for different grids	55

List of Figures

2.1	Flux vector Splitting for $(i, j)^{th}$ cell	21
2.2	Flux Estimation for the cell	26
2.3	Third order interpolation within the cells	27
2.4	Cell distribution in 1-D Flow	30
2.5	Periodic Boundary Zone	34
2.6	Solid Boundary condition for Euler calculations	35
2.7	Solid Boundary condition for Navier-Stokes calculations	37
2.8	Normal Vector profile for the Solid Boundary Surface	38
2.9	$(l, m)^{th}$ cell	42
2.10	Phase error comparison for different schemes	46
2.11	Numerical dissipation comparison for different schemes	46
3.1	Transformed Plane of a domain	52
3.2	Straight Cascade Profile	57
3.3	Algebraic grid for NACA0012 straight cascade $\alpha = 0^0$ and $\beta = 0^0$. .	58
3.4	Elliptic grid for NACA0012 straight cascade $\alpha = 0^0$ and $\beta = 0^0$. . .	59
3.5	Algebraic grid for DCA cascade $\alpha = 139.5^0$ and $\beta = 0^0$	60
3.6	Elliptic grid for DCA cascade $\alpha = 139.5^0$ and $\beta = 0^0$	61

4.1	Pressure Variation for NACA0012 Euler Solution <i>Contⁿ</i> . Mach No.= 0.8, CFL= 10^{-3}	69
4.2	Pressure Variation for NACA0012 Euler Solution. Mach No.= 0.8, CFL= 10^{-3}	70
4.3	Pressure Variation for NACA0012 Navier-Stokes Solution <i>Contⁿ</i> . Mach No.= 0.8, CFL= 10^{-5}	71
4.4	Pressure Variation for NACA0012 Navier-Stokes Solution <i>Contⁿ</i> . Mach No.= 0.8, CFL= 10^{-5}	72
4.5	Pressure Variation for NACA0012 Navier-Stokes Solution <i>Contⁿ</i> . Mach No.= 0.8, CFL= 10^{-5}	73
4.6	Pressure Variation for NACA0012 Navier-Stokes Solution <i>Contⁿ</i> . Mach No.= 0.8, CFL= 10^{-5}	74
4.7	Pressure Variation for NACA0012 Navier-Stokes Solution <i>Contⁿ</i> . Mach No.= 0.8, CFL= 10^{-5}	75
4.8	Pressure Variation for NACA0012 Navier-Stokes Solution <i>Contⁿ</i> . Mach No.= 0.8, CFL= 10^{-5}	76
4.9	Pressure Variation for NACA0012 Navier-Stokes Solution. Mach No.= 0.8, CFL= 10^{-5}	77
4.10	Pressure Variation for DCA Navier-Stokes Solution <i>Contⁿ</i> . Mach No.=0.3, CFL= 10^{-4}	78
4.11	Pressure Variation for DCA Navier-Stokes Solution <i>Contⁿ</i> . Mach No.=0.3, CFL= 10^{-4}	79
4.12	Pressure Variation for DCA Navier-Stokes Solution <i>Contⁿ</i> . Mach No.=0.3, CFL= 10^{-4}	80
4.13	Pressure Variation for DCA Navier-Stokes Solution <i>Contⁿ</i> . Mach No.=0.3, CFL= 10^{-4}	81

4.14 Pressure Variation for DCA Navier-Stokes Solution <i>Contⁿ</i> . Mach No.=0.3, CFL= 10^{-4}	82
4.15 Pressure Variation for DCA Navier-Stokes Solution <i>Contⁿ</i> . Mach No.=0.3, CFL= 10^{-4}	83
4.16 Pressure Variation for DCA Navier-Stokes Solution <i>Contⁿ</i> . Mach No.=0.3, CFL= 10^{-4}	84
4.17 Pressure Variation for DCA Navier-Stokes Solution. Mach No.=0.3, CFL= 10^{-4}	85

Abstract

In the field of Turbomachinery, apart from design point of view, viscous flow predictions carry greater importance for the analysis of off-design conditions; which would be possible by computing unsteady compressible Navier-Stokes equation. In the present work, viscous flow past cascades have been studied.

An unsteady two-dimensional viscous compressible Navier-Stokes equation is being solved in this context by Finite Volume(FV) principle using cell centered approach. A higher order upwinding scheme (MUSCL interpolation) along with Steger and Warming Flux Vector Splitting has been implemented. A time-step marching is carried out with Rational Runge-Kutta(RRK) time integration method. Elliptically generated grid is used to compute both Euler and Navier-Stokes equation.

The flow has been analysed for NACA0012 as well as DCA cascade for various different inflow Mach numbers in subsonic and transonic condition. Finally, various types of finite volume schemes have been theoretically analysed by spectral method and their efficacy have been discussed.

Chapter 1

Introduction

1.1 Role of CFD in Turbomachinery

The Physical aspects of any fluid flow are governed by three fundamental principles based on conservation of mass, momentum and energy. These fundamental principles can be expressed in terms of mathematical equations, which in their most general form are usually partial differential equations. Computational Fluid Dynamics (CFD) is the science of determining a numerical solution to these governing equations of fluid flow whilst advancing the solution through space or time to obtain a numerical description of the complete flow field of interest.

The governing equations in fluid dynamics, the unsteady Navier-Stokes equations, have been known for over a century and half. However, the analytical investigation of reduced forms of these equations is still an active area of research as is the problem of turbulence closure for the averaged form of the equations.

Experimental fluid dynamics plays an important role in validating and de-

lineating the limits of the various approximations to the governing equations. The wind tunnel, for example, as a piece of experimental equipment, provides an effective means of simulating real flows. Traditionally this has provided a cost effective alternative to full scale measurement. However, in the design of equipment that depends critically on the flow behaviour, for example the aerodynamic design of an aircraft, full scale measurement as part of the design process is not practical and sometimes not feasible. This situation has led to an increasing interest in the development of a numerical wind tunnel.

The role of CFD in engineering predictions has become so strong that today it may be viewed as a new third dimension of fluid dynamics, the other two dimensions being the above stated classical cases of pure experiment and pure theory.

The development of more powerful computers have furthered the advances being made in the field of computational fluid dynamics. Consequently CFD is now the preferred means of testing alternative designs in many engineering companies before final experimental testing takes place.

1.2 Aim of the Thesis

A computational study utilizing higher order upwinding scheme is being proposed to solve two dimensional unsteady viscous compressible Navier-Stokes equation for flow past cascade. The aim is to study the effect of gust on the performance of the cascade. Here the gust is unsteadiness of the oncoming flow which is due to the shed wake of the previous stage.

Thus the computational study of such a complex flow phenomena would require solving the three dimensional unsteady Navier-Stokes equation. Here in this project, we are going to study the two dimensional unsteady compressible flow past

cascade. The understanding and prediction of such viscous flows are of great interest and importance in turbomachinery design and analysis.

Here for the Navier-Stokes solver we have used the *flux vector splitting method* utilising *higher order upwinding in a finite volume formulation*. The higher order upwinding is based on *MUSCL interpolation* and can be used for analysis of aeroacoustic problem in cascade as well.

1.3 Thesis Overview

For the basics of unsteady compressible flows and fundamentals of CFD, we have followed Fletcher [1], Hirsch [9], Anderson [15], Courant & Friedrichs [16], White [17] and David [18].

The development of theoretical methods to predict unsteady flows in turbomachinery is a difficult task and determining the time-dependent two or three dimensional flows of a viscous compressible fluid through a geometric configuration is of enormous complexity. An unsteady viscous phenomenon in case of a turbomachinery include intricate shock patterns, viscous separation and many other physical problems. Because of the complexity of the unsteady phenomena occurring within the machine, design and development approaches have been based generally on steady-state theoretical aerodynamics along with the empirical correlations to account for unsteady effects.

For unsteady viscous compressible flows, explicit schemes are more accurate in predicting the flow solution as that of implicit schemes. For numerical stability, however, the time-integration step size used in integration must be fairly small, while allowable time step is governed by CFL number. In this thesis we have used these aspects to solve unsteady viscous two dimensional compressible Navier-Stokes

equation for flow past cascade.

There are three major components of this thesis problem. First and foremost is the grid generation, which is needed to study flow past a complex body. Here, grid generation has been carried out by two methods: First is an Algebraic and second is an Elliptic grid generation method. The topic grid generation is very well described by Fletcher [1]. Thompson et al [2] give details about an elliptic grid generation technique for internal as well as external flow domain and this has been followed in this thesis.

The second and the most important component is the Navier-Stokes solver. The formulation and flux vector splitting ideas are well described in Shu et al [3] and Hirsch [9]. The flux vector splitting scheme given by Steger and Warming [4] gives better efficiency in calculations of eigenvalues and eigenvectors while constructing the split fluxes. This flux vector splitting has been implemented in the solver. Multidimensional splittings given by Deconinck [19] give the mathematical approach of the splitting schemes. The MUSCL interpolation scheme during discretization has been used for better accuracy, which is also given in Hirsch [9]. For time-step marching, the Rational Runge Kutta method by Morinishi and Satofuka [10] is being used. The boundary conditions are implemented as given in Ravichandran [11] and Mandal [7]. Finally, the viscous terms calculation and full Navier-Stokes solution is carried out following Shu et al [3] and Kallinderis & Baron [13]. For the computations, geometries are given in Fottener [6]. For NACA0012 straight cascade, the pressure variation along top and bottom surfaces and Mach number distribution has been compared with Mandal [7]. While, for the DCA cascades, the experimental results given in Fottener [6] for the pressure coefficients on surface, Mach number variation and the velocity profile close to aerofoils have been compared with computed solutions.

Also, we have performed the spectral analysis of flux vector splitting finite volume method for compressible flows in general. Sengupta and Sengupta [14] gives the details about spectral analysis for an incompressible flows which has been extended here. We have analysed various flux vector splitting schemes, for their phase errors and numerical dissipation, which are given in Hirsch [9].

Chapter 2

Governing Equations for Compressible Flows

2.1 Introduction

Most compressible flows of practical importance that require consideration of the full Navier-Stokes equation are turbulent. The external compressible flows at large Reynolds numbers, viscous and turbulence effects are only significant close to solid surface in the absence of massive separation. Consequently, for such flows occasionally dissipative terms associated with the direction normal to the surface are as Thin Layer Navier-Stokes (TLNS).

However, for the transonic flow past cascades, the complicated shock boundary layer interaction produce local region of separated flows and it is appropriate to use full Navier-Stokes equation to capture the phenomenon accurately. If the flow is locally supersonic, embedded shock waves are likely to be present. If these

weak, accurate solutions can be usually obtained without modification to the basic computational algorithm, other than the inclusion of additional numerical dissipation.

This thesis discusses a Finite Volume Method(FVM) for discretisation of full Navier-Stokes equation. Finite Volume Method(FVM) is called as local method since the discretised equations are sparse. The finite volume method has additional advantage of discretising directly the conservative form of governing equation. This implies that the discretised equation preserves the conservation laws, reducing numerical errors.

The Flux Vector Splitting(FVS) scheme, that has been used here, can be applied to the conservation law form which allows the shock waves to be captured as weak solutions to the governing equation and circumvents the difficulty of applying shock fitting techniques to arbitrary flows.

Since the fluxes and geometric terms are evaluated only at the interfaces, the higher order MUSCL-type interpolation in space plays a major role in solution accuracy. Higher order Runge-Kutta methods for time integration similarly plays an important role in the selection of proper time step. A two-stage Rational Runge-Kutta(RRK) method has been implemented for the flow past cascade.

The application of boundary conditions is also crucial to have a correct solution, especially, for the internal transonic flows. We have used characteristics based boundary conditions.

2.2 Finite Volume Formulation for Compressible Flow Equations

2.2.1 Introduction

The computation of inviscid or viscous compressible flow past cascade is difficult because of the presence of shock with the strong gradients of finite thickness apart from the shear layer close to the body. This shock thickness can still be very thin to be resolved by the used grid and may lead to non-linear numerical instability. The standard techniques to obviate the difficulty are either to add explicit artificial dissipation or to employ an upwind biased scheme with implicit built-in dissipation. In the present effort the latter has been used via the MUSCL interpolation of variables.

The conservative formulation for the Navier-Stokes equation in Cartesian coordinate system is given by,

$$\frac{\partial \vec{q}}{\partial t} + \frac{\partial \vec{F}_c}{\partial x} + \frac{\partial \vec{G}_c}{\partial y} = \frac{1}{Re} \left(\frac{\partial \vec{F}_v}{\partial x} + \frac{\partial \vec{G}_v}{\partial y} \right) \quad (2.1)$$

where, \vec{F}_c and \vec{G}_c are the convective terms in 2-D flow domain while \vec{F}_v and \vec{G}_v are the dissipative or viscous terms in the flow domain.

This Navier-Stokes equation has been solved for the flow variables: density (ρ), velocities $[u, v]^T$, pressure (p) and the total energy (E).

The total energy (E) could be related to pressure as,

$$p = (\gamma - 1)[E - \frac{1}{2}\rho(u^2 + v^2)]$$

The local speed of sound can be expressed as,

$$c^2 = \frac{\gamma p}{\rho}$$

where γ is the ratio of the specific heats C_p and C_v .

The unknown vector \vec{q} is represented by,

$$\vec{q} = [\rho, \rho u, \rho v, E]^T$$

To enable us to treat non-uniform grids or mapping a complex physical domain, we have worked in the transformed plane (ξ, η) such that

$$\xi = \xi(x, y)$$

$$\eta = \eta(x, y)$$

The main reason for working in the transformed plane is due to our strategy of using higher order spatial interpolation for accuracy.

The governing equation (2.1) therefore transforms to,

$$\frac{\partial \hat{q}}{\partial t} + \frac{\partial \hat{F}_c}{\partial \xi} + \frac{\partial \hat{G}_c}{\partial \eta} = \frac{1}{Re} \left(\frac{\partial \hat{F}_v}{\partial \xi} + \frac{\partial \hat{G}_v}{\partial \eta} \right) \quad (2.2)$$

where,

$$\hat{q} = \vec{q}/J,$$

$$\hat{F}_c = U\vec{q} + p[0 \ \xi_x \ \xi_y \ U]^T/J,$$

$$\hat{G}_c = V\vec{q} + p[0 \ \eta_x \ \eta_y \ V]^T/J,$$

$$\hat{F}_v = (\xi_x \vec{r} + \xi_y \vec{s})/J,$$

$$\hat{G}_v = (\eta_x \vec{r} + \eta_y \vec{s})/J,$$

$$\vec{r} = \mu(0 \ \tau_{11} \ \tau_{21} \ \sigma_1)^T,$$

$$\vec{s} = \mu(0 \ \tau_{12} \ \tau_{22} \ \sigma_2)^T$$

with

$$\tau_{11} = \frac{4}{3}(\xi_x u_\xi + \eta_x u_\eta) - \frac{2}{3}(\xi_y v_\xi + \eta_y v_\eta),$$

$$\tau_{21} = \tau_{12} = \xi_y u_\xi + \eta_y u_\eta + \xi_x v_\xi + \eta_x v_\eta,$$

$$\begin{aligned}\tau_{22} &= \frac{4}{3}(\xi_y v_\xi + \eta_y v_\eta) - \frac{2}{3}(\xi_x u_\xi + \eta_x u_\eta), \\ \sigma_1 &= u\tau_{11} + v\tau_{12} + \frac{\kappa}{(\gamma - 1)P_r}[\xi_x(c^2)_\xi + \eta_x(c^2)_\eta], \\ \sigma_2 &= u\tau_{21} + v\tau_{22} + \frac{\kappa}{(\gamma - 1)P_r}[\xi_y(c^2)_\xi + \eta_y(c^2)_\eta]\end{aligned}$$

and the Jacobian of transformation and the contravariant component of the velocity in the transformed plane are given by,

$$J = \xi_x \eta_y - \xi_y \eta_x,$$

$$U = \xi_x u + \xi_y v,$$

$$V = \eta_x u + \eta_y v$$

Here in the equation (2.2), Re is the Reynolds number, τ'_{ij} s are the shear stress terms and σ'_i s are the loss terms in the energy equation due to viscous terms.

Enthalpy can be calculated as,

$$H = \frac{E + p}{\rho}$$

Prandtl number is given by,

$$P_r = \frac{\mu C_p}{\kappa}$$

where μ is the viscosity which can be obtained by Sutherland's [16] Law as,

$$\mu = 1.45 * 10^{-6} * \frac{T^{3/2}}{T + 110}$$

with T is the local temperature specified, C_p is the specific heat coefficient at constant pressure and κ is the coefficient of thermal conductivity.

The Navier-Stokes equation can be reduced to Euler equation when $Re \rightarrow \infty$, i.e. by retaining convective terms while neglecting the viscous terms. The Euler equation is,

$$\frac{\partial \hat{q}}{\partial t} + \frac{\partial \hat{F}_c}{\partial \xi} + \frac{\partial \hat{G}_c}{\partial \eta} = 0 \quad (2.3)$$

The finite volume formulation for the equation (2.2) in terms of explicit time marching scheme for any arbitrary $(i, j)^{th}$ cell can be shown as,

$$\frac{\hat{q}^{n+1} - \hat{q}^n}{\partial t} S_{i,j} - \oint (\hat{F}_c d\eta - \hat{G}_c d\xi) + \frac{1}{Re} \oint (\hat{F}_v d\eta - \hat{G}_v d\xi) = 0 \quad (2.4)$$

which gives,

$$\hat{q}^{n+1} = \hat{q}^n - \frac{\Delta t}{S_{i,j}} \oint (\hat{F}_c d\eta - \hat{G}_c d\xi) + \frac{1}{Re} \oint (\hat{F}_v d\eta - \hat{G}_v d\xi) \quad (2.5)$$

where the superscripts n and $n + 1$ show the present and future time step values of fluxes at any $(i, j)^{th}$ cell.

The time Δt represents the value of time step marching while $S_{i,j}$ is the area of $(i, j)^{th}$ cell over which we are solving the Navier-Stokes equation.

Here, the variables are referred to the cell center values. The integral represents the flux integral performed over the cell interfaces.

2.2.2 Nondimensionalization of the Quantities in Flow Equations

Nondimensionalization reduces the number of computations to a single parameter value. In arriving at equation (2.2) the following nondimensionalization has been used,

Following scales have been used for nondimensionalization of the flow variables:

1. Reference Velocity: U_∞
2. Reference free-stream properties: $\rho_\infty, u_\infty, v_\infty, E_\infty, M_\infty, c_\infty, H_\infty$ and p_∞ .
3. Reference Specific heats and universal gas constant: C_{p0}, C_{v0}, R_0 .
4. Other reference parameters: L, μ_0, T_0, κ_0

For the computational cascade domain, nondimensionalised parameters can be calculated as given below. The asterix (*) quantities are the dimensional quantities while nondimensionalised quantities are written without any super or subscript. The Tangential Velocity is calculated as,

$$u = u^*/U_\infty$$

The Normal Velocity is calculated as,

$$v = v^*/U_\infty$$

The Density is given by,

$$\rho = \rho^*/\rho_\infty$$

The Energy can be shown as,

$$E = \frac{E^*}{\rho_\infty U_\infty^2}$$

The Enthalpy is shown as,

$$H = \frac{H^*}{\rho_\infty U_\infty^2}$$

The Speed of Sound can be shown as,

$$c = c^*/c_\infty$$

Using these nondimensionalized quantities we can express nondimensionalized viscosity by Sutherland's [16] law as,

$$\mu = \frac{\mu^*}{\mu_0} = \left(\frac{T^*}{T_0}\right)^{3/2} * \left(\frac{T_0 + 110}{T^* + 110}\right)$$

Using these parameters one can proceed for the calculation of Prandtl number and many other related parameters.

2.2.3 Nondimensionalizing the Inflow Boundary Conditions

The computational domain with the specified boundaries has been shown in Figure (3.2).

For the free-stream boundary condition the reference parameters are $\rho_\infty, u_\infty, v_\infty$, and E_∞ .

For the inflow of the cascade domain, density is

$$\rho = 1 \text{ as } \rho^* = \rho_\infty.$$

The velocities are,

$$u = \cos(\theta) \text{ and } v = \sin(\theta)$$

as $u^* = U_\infty, v^* = U_\infty$ and θ is the angle at which the flow enters the flow domain.

The pressure can be shown as,

$$p = 1 \text{ as } p^* = p_\infty.$$

Hence, the free-stream energy can be calculated using nondimensionalized quantities as,

$$E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho (u^2 + v^2)$$

reduce to,

$$E = \frac{1}{\gamma - 1} + \frac{1}{2}$$

This shows that by nondimensionalization of the various parameters, we are making equation flexible for any large or small physical values.

2.3 Flux Vector Splitting Scheme

2.3.1 Introduction

All second-order central schemes generates oscillations which have to be damped by the addition of artificial dissipation terms to suppress the nonlinear instabilities.

In smooth regions of the flow, where the flow variables can be considered as continuous, central schemes based on Taylor series expansion can be used with any order of accuracy. This is valid for supersonic flows also, where the apparent contradiction between the physical one-way propagation of waves and the symmetrical central difference schemes are direction independent, being resolved by considering the analytic continuation properties of smooth function. The validity of Taylor series expansions express, indeed, a most remarkable property of continuous functions, namely that suffices the value of a continuous function in a *single* cell to be able to reconstruct the function in an increasingly large domain around that cell. In the limit, the complete *local* knowledge of its value and all its derivatives in a single cell, is equivalent to the knowledge of function everywhere. However, as soon as discontinuities appear this information is destroyed and more physical input is required in order to resolve the non-linear behaviour.

The first level introduces only information on the sign of the eigenvalues, whereby the flux terms are split and discretized directionally according to the sign of the associated propagation speeds. This leads to the *flux vector splitting* methods, one of which we are using for our case of study in this thesis.

2.3.2 Multi-Dimensional Flux Vector Splitting

The flux vector splitting upwind methods can be extended to multi-dimensional flow by applying the one-dimensional splitting to each flux components separately

according to the sign of the associated eigenvalues.

The Euler equation given in the section 2.2 for the transformed plane can be shown as.

$$\frac{\partial \hat{q}}{\partial t} + \frac{\partial \hat{F}_c}{\partial \xi} + \frac{\partial \hat{G}_c}{\partial \eta} = 0 \quad (2.6)$$

The convective terms from the equation (2.6) can be rewritten as,

$$\frac{\partial \hat{F}_c}{\partial \xi} = \frac{\partial \hat{F}_c}{\partial \hat{q}} \frac{\partial \hat{q}}{\partial \xi} = A \frac{\partial \hat{q}}{\partial \xi} \quad (2.7)$$

and

$$\frac{\partial \hat{G}_c}{\partial \eta} = \frac{\partial \hat{G}_c}{\partial \hat{q}} \frac{\partial \hat{q}}{\partial \eta} = B \frac{\partial \hat{q}}{\partial \eta} \quad (2.8)$$

where,

$$A = \frac{\partial \hat{F}_c}{\partial \hat{q}} \quad \text{and} \quad B = \frac{\partial \hat{G}_c}{\partial \hat{q}}$$

For the equations (2.7) and (2.8) the matrices A and B, which are the Jacobian Flux Matrices [3] are given as,

$$A = \begin{pmatrix} 0 & \xi_x & \xi_y & 0 \\ \frac{\gamma-1}{2} \xi_x q_1^2 - uU & U + (2-\gamma)u\xi_x & u\xi_y - (\gamma-1)\xi_x v & (\gamma-1)\xi_x \\ \frac{\gamma-1}{2} \xi_y q_1^2 - vU & v\xi_x - (\gamma-1)\xi_y u & U + (2-\gamma)v\xi_y & (\gamma-1)\xi_y \\ (\gamma-1)U q_1^2 - \frac{\gamma U e}{\rho} & \frac{\gamma e}{\rho} \xi_x - (uU + \frac{\xi_x}{2} q_1^2)(\gamma-1) & \frac{\gamma e}{\rho} \xi_y - (vU + \frac{\xi_y}{2} q_1^2)(\gamma-1) & \gamma U \end{pmatrix}$$

and

$$B = \begin{pmatrix} 0 & \eta_x & \eta_y & 0 \\ \frac{\gamma-1}{2} \eta_x q_1^2 - uV & V + (2-\gamma)u\eta_x & u\eta_y - (\gamma-1)\eta_x v & (\gamma-1)\eta_x \\ \frac{\gamma-1}{2} \eta_y q_1^2 - vV & v\eta_x - (\gamma-1)\eta_y u & V + (2-\gamma)v\eta_y & (\gamma-1)\eta_y \\ (\gamma-1)V q_1^2 - \frac{\gamma V e}{\rho} & \frac{\gamma e}{\rho} \eta_x - (uV + \frac{\eta_x}{2} q_1^2)(\gamma-1) & \frac{\gamma e}{\rho} \eta_y - (vV + \frac{\eta_y}{2} q_1^2)(\gamma-1) & \gamma V \end{pmatrix}$$

The Jacobian matrices A and B have four eigenvalues which can be represented in terms of right and left eigenvectors (R and S) as,

$$\Lambda_A = R A R^{-1} = \begin{vmatrix} U - c\sqrt{\xi_x^2 + \xi_y^2} & 0 & 0 & 0 \\ 0 & U & 0 & 0 \\ 0 & 0 & U & 0 \\ 0 & 0 & 0 & U + c\sqrt{\xi_x^2 + \xi_y^2} \end{vmatrix} \quad (2.9)$$

and

$$\Lambda_B = S B S^{-1} = \begin{vmatrix} V - c\sqrt{\eta_x^2 + \eta_y^2} & 0 & 0 & 0 \\ 0 & V & 0 & 0 \\ 0 & 0 & V & 0 \\ 0 & 0 & 0 & V + c\sqrt{\eta_x^2 + \eta_y^2} \end{vmatrix} \quad (2.10)$$

where, R and R^{-1} are the right and left eigenvectors whereas S and S^{-1} are the corresponding eigenvectors for B-matrix.

These eigenvectors can be shown as,

$$R = \begin{pmatrix} 1 & 0 & 1 & 1 \\ u - \tilde{\xi}_x c & -\tilde{\xi}_y & u & u + \tilde{\xi}_x c \\ v - \tilde{\xi}_y c & \tilde{\xi}_x & v & v - \tilde{\xi}_y c \\ H - \tilde{\Theta}_\xi c & -\tilde{\xi}_y u + \tilde{\xi}_x v & \frac{1}{2}(u^2 + v^2) & H + \tilde{\Theta}_\xi c \end{pmatrix}, \quad (2.11)$$

$$R^{-1} = \frac{1}{2} \begin{pmatrix} (b_2 + \frac{\Theta_\xi}{c}) & -(b_1 u + \frac{\tilde{\xi}_x}{c}) & -(b_1 v + \frac{\tilde{\xi}_y}{c}) & b_1 \\ 2(\tilde{\xi}_y u - \tilde{\xi}_x v) & -2\tilde{\xi}_y & 2\tilde{\xi}_x & 0 \\ 2(1 - b_2) & 2b_1 u & 2b_1 v & -2b_1 \\ (b_2 - \frac{\Theta_\xi}{c}) & -(b_1 u - \frac{\tilde{\xi}_x}{c}) & -(b_1 v - \frac{\tilde{\xi}_y}{c}) & b_1 \end{pmatrix} \quad (2.12)$$

and similarly,

$$S = \begin{pmatrix} 1 & 1 & 0 & 1 \\ u - \tilde{\eta}_x c & u & \tilde{\eta}_y & u + \tilde{\eta}_x c \\ v - \tilde{\eta}_y c & v & -\tilde{\eta}_x & v + \tilde{\eta}_y c \\ H - \tilde{\Theta}_\eta c & \frac{1}{2}(u^2 + v^2) & \tilde{\eta}_y u - \tilde{\eta}_x v & H + \tilde{\Theta}_\eta c \end{pmatrix}, \quad (2.13)$$

$$S^{-1} = \frac{1}{2} \begin{pmatrix} (b_2 + \frac{\Theta_\eta}{c}) & -(b_1 u + \frac{\tilde{\eta}_x}{c}) & -(b_1 v - \frac{\tilde{\eta}_y}{c}) & b_1 \\ 2(1 - b_2) & 2b_1 u & 2b_1 v & -2b_1 \\ 2(\tilde{\eta}_x v - \tilde{\eta}_y u) & 2\tilde{\eta}_y & -2\tilde{\eta}_x & 0 \\ (b_2 - \frac{\Theta_\eta}{c}) & -(b_1 u - \frac{\tilde{\eta}_x}{c}) & -(b_1 v - \frac{\tilde{\eta}_y}{c}) & b_1 \end{pmatrix} \quad (2.14)$$

where,

$$b_1 = \frac{\gamma - 1}{c^2},$$

$$b_2 = \frac{1}{2}(u^2 + v^2) * b_1,$$

$$\tilde{\xi}_x = \frac{\xi_x}{\sqrt{\xi_x^2 + \xi_y^2}},$$

$$\tilde{\xi}_y = \frac{\xi_y}{\sqrt{\xi_x^2 + \xi_y^2}},$$

$$\tilde{\Theta}_\xi = \tilde{\xi}_x u + \tilde{\xi}_y v$$

and

$$\tilde{\eta}_x = \frac{\eta_x}{\sqrt{\eta_x^2 + \eta_y^2}},$$

$$\tilde{\eta}_y = \frac{\eta_y}{\sqrt{\eta_x^2 + \eta_y^2}},$$

$$\tilde{\Theta}_\eta = \tilde{\eta}_x u + \tilde{\eta}_y v$$

Hence, an upwind formulation can be obtained with the Jacobian matrices as,

$$A^\pm = R\Lambda_A^\pm R^{-1} \quad (2.15)$$

and

$$B^\pm = S\Lambda_B^\pm S^{-1} \quad (2.16)$$

with the Jacobian matrix property of,

$$A = A^+ + A^- \quad (2.17)$$

and

$$| A | = A^+ - A^- \quad (2.18)$$

similarly,

$$B = B^+ + B^- \quad (2.19)$$

and

$$| B | = B^+ - B^- \quad (2.20)$$

The fluxes associated with these split Jacobian matrices are obtained from the remarkable property of homogeneity of the flux vector $\hat{F}_c(\hat{q})$. The function $\hat{F}_c(\hat{q})$ is a homogeneous function of degree one of \hat{q} .

Hence the flux splitting can be defined as,

$$\hat{F}_c = A\hat{q} = \hat{F}_c^+ + \hat{F}_c^-$$

and

$$\hat{G}_c = B\hat{q} = \hat{G}_c^+ + \hat{G}_c^-$$

where

$$\hat{F}_c^+ = A^+ \hat{q}^+, \quad \hat{F}_c^- = A^- \hat{q}^- \quad (2.21)$$

and

$$\hat{G}_c^+ = B^+ \hat{q}^+, \quad \hat{G}_c^- = B^- \hat{q}^- \quad (2.22)$$

The splitting of flux vectors as shown in equations (2.21) and (2.22) are directed by the property of information propagation that can be achieved by upwind schemes.

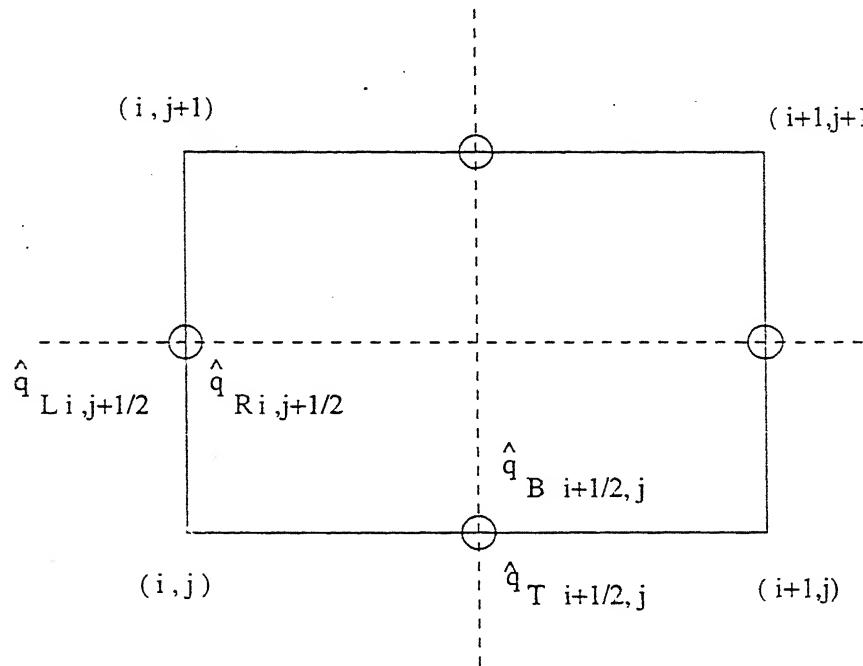
The equations (2.21) and (2.22) show, A^+ as the Jacobian matrix corresponding to the disturbances propagating from left to right and A^- corresponds to the disturbances propagating from right to left. Similarly, B^+ corresponds to disturbances propagating from bottom to top while B^- from top to bottom.

In this form of flux vector splitting \hat{q}^+ is the flow property just to the left of the surface considered while \hat{q}^- is just to the right of the surface considered for equation (2.21), while for the equation (2.22), \hat{q}^+ is just to the top of the surface considered and \hat{q}^- is to the bottom of the surface considered.

In actual computer coding for our convenience, we are assuming these split flux quantities as, \hat{q}_L, \hat{q}_R for left and right side while \hat{q}_T, \hat{q}_B for top and bottom side of the surface considered.

The Figure (2.1) shows the flux vector splitting for $(i, j)^{th}$ cell for the left and bottom surface of the cell considered.

The eigenvalue matrices shown by equation (2.9) and (2.10) decides the flow nature i.e. subsonic or supersonic at any cell in physical domain. The flux splitting technique by Steger and Warming [4] that has been used here is explained in next subsection.

Figure 2.1: Flux vector Splitting for $(i, j)^{th}$ cell

2.3.3 Steger and Warming Flux Vector Splitting

From the equation (2.21) and (2.22), we can calculate the eigenvalues (Λ) for A^+ , A^- and B^+ , B^- matrices by the method suggested by Steger and Warming [4] as,

$$\Lambda^+ = \frac{\Lambda^+ + |\Lambda^+|}{2} \quad (2.23)$$

$$\Lambda^- = \frac{\Lambda^- - |\Lambda^-|}{2} \quad (2.24)$$

For each of the eigenvalues of Λ

$$\Lambda_A^\pm = \begin{vmatrix} \lambda_1^\pm & 0 & 0 & 0 \\ 0 & \lambda_2^\pm & 0 & 0 \\ 0 & 0 & \lambda_3^\pm & 0 \\ 0 & 0 & 0 & \lambda_4^\pm \end{vmatrix} \quad (2.25)$$

where,

$$\lambda_1^\pm = \frac{(U + c\sqrt{\xi_x^2 + \xi_y^2}) \pm |(U + c\sqrt{\xi_x^2 + \xi_y^2})|}{2},$$

$$\lambda_2^\pm = \frac{U \pm |U|}{2},$$

$$\lambda_3^\pm = \frac{U \pm |U|}{2},$$

$$\lambda_4^\pm = \frac{(U - c\sqrt{\xi_x^2 + \xi_y^2}) \pm |(U - c\sqrt{\xi_x^2 + \xi_y^2})|}{2}$$

and

$$\Lambda_B^\pm = \begin{vmatrix} \Delta_1^\pm & 0 & 0 & 0 \\ 0 & \Delta_2^\pm & 0 & 0 \\ 0 & 0 & \Delta_3^\pm & 0 \\ 0 & 0 & 0 & \Delta_4^\pm \end{vmatrix} \quad (2.26)$$

where,

$$\Delta_1^\pm = \frac{(V + c\sqrt{\eta_x^2 + \eta_y^2}) \pm |(V + c\sqrt{\eta_x^2 + \eta_y^2})|}{2},$$

$$\Delta_2^\pm = \frac{V \pm |V|}{2},$$

$$\Delta_3^\pm = \frac{V \pm |V|}{2},$$

$$\Delta_4^\pm = \frac{(V - c\sqrt{\eta_x^2 + \eta_y^2}) \pm |(V - c\sqrt{\eta_x^2 + \eta_y^2})|}{2}$$

Here, Λ^+ has only positive eigenvalues, Λ^- only negative values and such that,

$$\Lambda = \Lambda^+ + \Lambda^-, \quad |\Lambda| = \Lambda^+ - \Lambda^-$$

2.3.4 Properties of Split Flux Vectors

The most important concept of flux splitting is that, it is totally dependent on the fact that the fluxes are homogeneous function of degree one in \hat{q} . One can not therefore directly apply this approach to a general scalar flux function $f(q)$. Since the only homogeneous scalar flux function of degree one is the linear case $f = aq$.

As shown in equations (2.21) and (2.22) splitting of Jacobian matrices give,

$$A = A^+ + A^- \quad (2.27)$$

and

$$B = B^+ + B^- \quad (2.28)$$

with

$$\hat{F}_c = A\hat{q} = \hat{F}_c^+ + \hat{F}_c^- = A^+\hat{q} + A^-\hat{q} \quad (2.29)$$

$$\hat{G}_c = B\hat{q} = \hat{G}_c^+ + \hat{G}_c^- = B^+\hat{q} + B^-\hat{q} \quad (2.30)$$

and

$$\frac{\partial \hat{F}_c}{\partial \hat{q}} = A = \frac{\partial \hat{F}_c^+}{\partial \hat{q}} + \frac{\partial \hat{F}_c^-}{\partial \hat{q}} = A^+ + A^- \quad (2.31)$$

$$\frac{\partial \hat{G}_c}{\partial \hat{q}} = B = \frac{\partial \hat{G}_c^+}{\partial \hat{q}} + \frac{\partial \hat{G}_c^-}{\partial \hat{q}} = B^+ + B^- \quad (2.32)$$

but one does not generally have the equality of the split Jacobian i.e.

$$\frac{\partial \hat{F}_c^+}{\partial \hat{q}} \neq A^+, \quad \frac{\partial \hat{F}_c^-}{\partial \hat{q}} \neq A^- \quad (2.33)$$

or

$$\frac{\partial \hat{G}_c^+}{\partial \hat{q}} \neq B^+, \quad \frac{\partial \hat{G}_c^-}{\partial \hat{q}} \neq B^- \quad (2.34)$$

These matrices do not have the same set of eigenvalues. The eigenvalues of A^+ and A^- as well as B^+ and B^- are the positive and negative values of A and B, that is λ_k^\pm and Δ_k^\pm , but this is not true for $\frac{\partial \hat{F}_c^\pm}{\partial \hat{q}}$ and $\frac{\partial \hat{G}_c^\pm}{\partial \hat{q}}$.

The non-equality of $\frac{\partial \hat{F}_c^\pm}{\partial \hat{q}}$ and $\frac{\partial \hat{G}_c^\pm}{\partial \hat{q}}$ with A^\pm and B^\pm has very important relation with regard to the definition of upwind schemes based on the concept of flux vector splitting. Therefore, the conservative equation in split form can be shown as,

$$\frac{\partial \hat{q}}{\partial t} + \frac{\partial \hat{F}_c^+}{\partial \xi} + \frac{\partial \hat{F}_c^-}{\partial \eta} + \frac{\partial \hat{G}_c^+}{\partial \eta} + \frac{\partial \hat{G}_c^-}{\partial \eta} = 0 \quad (2.35)$$

which in terms of quasi-linear form can be expressed as,

$$\frac{\partial \hat{q}}{\partial t} + \frac{\partial \hat{F}_c^+}{\partial \hat{q}} \frac{\partial \hat{q}}{\partial \xi} + \frac{\partial \hat{F}_c^-}{\partial \hat{q}} \frac{\partial \hat{q}}{\partial \eta} + \frac{\partial \hat{G}_c^+}{\partial \hat{q}} \frac{\partial \hat{q}}{\partial \eta} + \frac{\partial \hat{G}_c^-}{\partial \hat{q}} \frac{\partial \hat{q}}{\partial \eta} = 0 \quad (2.36)$$

Alternatively, if we write the quasi-linear form first and then split the Jacobians then,

$$\frac{\partial \hat{q}}{\partial t} + A \frac{\partial \hat{q}}{\partial \xi} + B \frac{\partial \hat{q}}{\partial \eta} = \frac{\partial \hat{q}}{\partial t} + A^+ \frac{\partial \hat{q}}{\partial \xi} + A^- \frac{\partial \hat{q}}{\partial \xi} + B^+ \frac{\partial \hat{q}}{\partial \eta} + B^- \frac{\partial \hat{q}}{\partial \eta} = 0 \quad (2.37)$$

The two formulations are not identical, as a consequence of equations (2.34) and (2.33) showing that the two operations of splitting fluxes and introducing the quasi-linearization do not commute. Equation (2.37) keeps the same physical characteristics which are separated accordingly to their sign while equation (2.36) does not represent a decomposition of the physical characteristics, but is the only linearization of consistent with the conservative form of equation (2.36).

2.4 Interpolation Scheme

2.4.1 Introduction

The motivation behind the interpolation schemes is the hope that the introduction of physical propagation properties in the discretization will prevent the generation of oscillations in the numerical solutions. This is only partly fulfilled in the sense that for nonlinear equations, such as Euler equation oscillation free results can be obtained for weak stationary discontinuities. However, this is not a general property, since it can be shown theoretically that the linear second-order upwind schemes always generate oscillations. The concept of *monotonicity* was introduced by Godunov (1959) to achieve the oscillation-free solution.

As a piecewise numerical approximation is equivalent to a first-order spatial discretization, a linear approximation on each cell gives second-order spatial discretization; while a quadratic representation on each cell leads to third-order spatial discretization. Here in this thesis, we have used third order MUSCL type of spatial discretization which has built-in dissipation for solving full Navier-Stokes equation.

2.4.2 High-Order Upwind Scheme- MUSCL approach

Second-order spatial accuracy can be achieved by introducing more upwind points in the schemes. The state variables at the cell interface or at the cell centre are obtained from an extrapolation between the neighbouring cell averages. This method for the generation of second-order upwind schemes via *variable extrapolation* is referred as the MUSCL approach. MUSCL acronym stands for, *Monotone Upstream-centred Schemes for Conservation Laws*, as developed by Van Leer (1979). The time integration methods based on a separate time and space discretisation, such as multi-

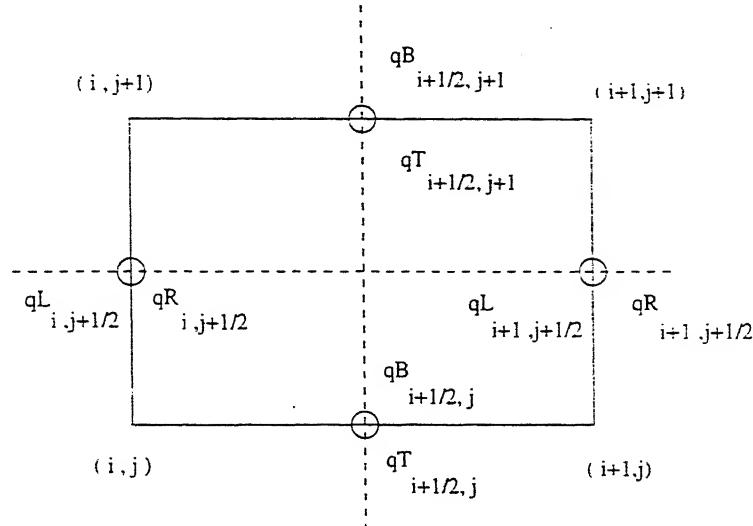


Figure 2.2: Flux Estimation for the cell

stage Runge-Kutta techniques or linear multistage methods (implicit schemes), can be directly applied to the MUSCL scheme.

The representative cell for the flux estimation is shown in Figure (2.2).

where, q is the flux quantity and the superscripts L and R refers to the left and right sides of a cell surface while, T and B are the top and bottom sides of the cell surface considered.

Higher-order upwinding scheme with the generalised formulas can be written as,

for the left hand side of the $(i, j)^{th}$ cell,

$$q^L_{i-1/2, j} = q_{i-1, j} + \frac{\epsilon}{4}[(1 - \kappa)(q_{i-1, j} - q_{i-2, j}) + (1 + \kappa)(q_{i, j} - q_{i-1, j})] \quad (2.38)$$

$$q^R_{i-1/2, j} = q_{i, j} - \frac{\epsilon}{4}[(1 + \kappa)(q_{i, j} - q_{i-1, j}) + (1 - \kappa)(q_{i-1, j} - q_{i, j})] \quad (2.39)$$

Similarly for the right hand side of the $(i, j)^{th}$ cell we have,

$$q^L_{i+1/2, j} = q_{i, j} + \frac{\epsilon}{4}[(1 - \kappa)(q_{i, j} - q_{i-1, j}) + (1 + \kappa)(q_{i+1, j} - q_{i, j})] \quad (2.40)$$

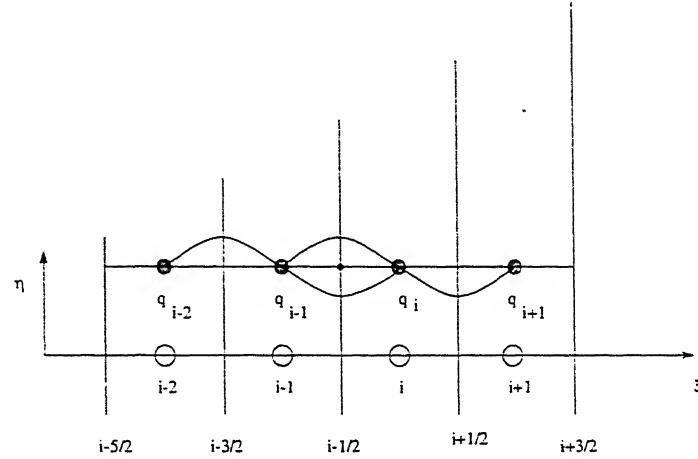


Figure 2.3: Third order interpolation within the cells

$$q_{i-1/2}^R = q_{i+1,j} - \frac{\epsilon}{4}[(1 + \kappa)(q_{i+1,j} - q_{i,j}) + (1 - \kappa)(q_{i+2,j} - q_{i+1,j})] \quad (2.41)$$

While, for top and bottom side, generalised interpolation formula is given by.

$$q_{i,j-1/2}^T = q_{i,j-1} + \frac{\epsilon}{4}[(1 - \kappa)(q_{i,j-1} - q_{i,j-2}) + (1 + \kappa)(q_{i,j} - q_{i,j-1})] \quad (2.42)$$

$$q_{i,j+1/2}^B = q_{i,j} - \frac{\epsilon}{4}[(1 + \kappa)(q_{i,j} - q_{i,j-1}) + (1 - \kappa)(q_{i,j+1} - q_{i,j})] \quad (2.43)$$

$$q_{i,j-1/2}^T = q_{i,j} + \frac{\epsilon}{4}[(1 - \kappa)(q_{i,j} - q_{i,j-1}) + (1 + \kappa)(q_{i,j+1} - q_{i,j})] \quad (2.44)$$

$$q_{i,j+1/2}^B = q_{i,j+1} - \frac{\epsilon}{4}[(1 + \kappa)(q_{i,j+1} - q_{i,j}) + (1 - \kappa)(q_{i,j+2} - q_{i,j+1})] \quad (2.45)$$

where, the introduction of parameter ϵ , such that $\epsilon=0$ is for a first order upwinding scheme and $\epsilon=1$ is for a higher order upwinding schemes. While κ defines the nature of the scheme given by Hirsch [9] as,

$\kappa = -1 \Rightarrow$ One sided linear interpolation scheme

$\kappa = 0 \Rightarrow$ Linear interpolation scheme

$\kappa = 1 \Rightarrow$ Central differencing scheme

$\kappa = 1/2 \Rightarrow$ QUICK scheme

and finally,

$\kappa = 1/3 \Rightarrow$ MUSCL interpolation scheme

In this thesis, we have used $\epsilon=1$ and $\kappa=1/3$ for third order MUSCL interpolation scheme.

From equation (2.38) to (2.45) substituting $\epsilon=1$ and $\kappa=1/3$ we get,

$$q_{i-1/2,j}^L = q_{i-1,j} + \frac{q_{i,j} - q_{i-1,j}}{3} + \frac{q_{i-1,j} - q_{i-2,j}}{6} \quad (2.46)$$

$$q_{i-1/2,j}^R = q_{i,j} + \frac{q_{i-1,j} - q_{i,j}}{3} + \frac{q_{i,j} - q_{i+1,j}}{6} \quad (2.47)$$

$$q_{i+1/2,j}^L = q_{i,j} + \frac{q_{i+1,j} - q_{i,j}}{3} + \frac{q_{i,j} - q_{i-1,j}}{6} \quad (2.48)$$

$$q_{i+1/2,j}^R = q_{i+1,j} + \frac{q_{i,j} - q_{i+1,j}}{3} + \frac{q_{i+1,j} - q_{i+2,j}}{6} \quad (2.49)$$

$$q_{i,j-1/2}^T = q_{i,j-1} + \frac{q_{i,j} - q_{i,j-1}}{3} + \frac{q_{i,j-1} - q_{i,j-2}}{6} \quad (2.50)$$

$$q_{i,j-1/2}^B = q_{i,j} + \frac{q_{i,j-1} - q_{i,j}}{3} + \frac{q_{i,j} - q_{i,j+1}}{6} \quad (2.51)$$

$$q_{i,j+1/2}^T = q_{i,j} + \frac{q_{i,j+1} - q_{i,j}}{3} + \frac{q_{i,j} - q_{i,j-1}}{6} \quad (2.52)$$

$$q_{i,j+1/2}^B = q_{i,j+1} + \frac{q_{i,j} - q_{i,j+1}}{3} + \frac{q_{i,j+1} - q_{i,j+2}}{6} \quad (2.53)$$

2.4.3 Free Stream Preservation for the Flux Calculations

The Euler equation for 2-D compressible, unsteady and uniform flow is rewritten for the transform plane as,

$$\frac{\partial \hat{q}}{\partial t} + \frac{\partial \hat{F}_c}{\partial \xi} + \frac{\partial \hat{G}_c}{\partial \eta} = 0 \quad (2.54)$$

For an uniform flow we have,

$$\vec{q}_1 = k_1, \quad \vec{q}_2 = k_2, \quad \vec{q}_3 = k_3, \quad \vec{q}_4 = k_4 \quad (2.55)$$

$$\vec{k} = [\rho_\infty, \rho_\infty u_\infty, \rho_\infty v_\infty, E_\infty] \quad (2.56)$$

where. ∞ is the free stream value of the flow variable.

From equations (2.55) and (2.56) we have,

$$\hat{q}_1 = k_1/J, \hat{q}_2 = k_2/J, \hat{q}_3 = k_3/J, \hat{q}_4 = k_4/J$$

where for a single $(i, j)^{th}$ cell Jacobian(J) will remain constant for all four quantities.

The convective terms from equation (2.54) can be rearranged as,

$$\frac{\partial \hat{F}_c}{\partial \xi} = A \frac{\partial \hat{q}}{\partial \xi} = A \frac{\partial}{\partial \xi} \left(\frac{\vec{k}}{J} \right) \quad (2.57)$$

similarly.

$$\frac{\partial \hat{G}_c}{\partial \eta} = B \frac{\partial \hat{q}}{\partial \eta} = B \frac{\partial}{\partial \eta} \left(\frac{\vec{k}}{J} \right) \quad (2.58)$$

$$\frac{\partial \hat{q}}{\partial t} = \frac{\partial}{\partial t} \left(\frac{\vec{k}}{J} \right) \quad (2.59)$$

From equations (2.57), (2.58) and (2.59) we can rearrange the 2-D Euler equation as,

$$\frac{\partial}{\partial t} \left(\frac{\vec{k}}{J} \right) + A \frac{\partial}{\partial \xi} \left(\frac{\vec{k}}{J} \right) + B \frac{\partial}{\partial \eta} \left(\frac{\vec{k}}{J} \right) = 0 \quad (2.60)$$

But $\frac{\partial}{\partial t} \left(\frac{\vec{k}}{J} \right) = 0$ as $J \neq J(t)$

Thus the equation (2.60) becomes,

$$A \frac{\partial}{\partial \xi} \left(\frac{\vec{k}}{J} \right) + B \frac{\partial}{\partial \eta} \left(\frac{\vec{k}}{J} \right) = 0 \quad (2.61)$$

In case of 1-D flow as shown in Figure (2.4),

$$\frac{\partial}{\partial \xi} \left(\frac{1}{J} \right) \Big|_i = \frac{(1/J)_{i+1} - (1/J)_{i-1}}{2\Delta \xi} = 0 \quad (2.62)$$

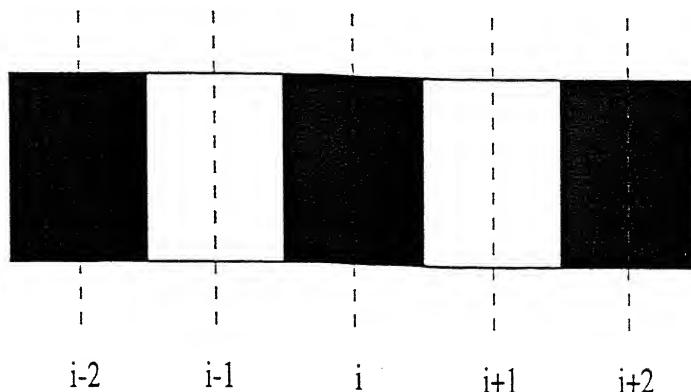


Figure 2.4: Cell distribution in 1-D Flow

which means,

$$J_{i+1} = J_{i-1} \quad (2.63)$$

The equation (2.62) tells us that, for uniform flow the jacobians are in checkerboard mode i.e. the alternative cell Jacobians must be same for the free stream preservation which is difficult to have in practice.

Similarly for 2-D flow equation we have checkerboard mode in ξ as well as in η direction. This shows that the grid plays major role in compressible flow calculations.

2.5 Time Integration Scheme

2.5.1 Introduction

To obtain a steady state solution for the flows at high Reynolds number, the time-step restriction becomes a severe impediment to overall efficiency. This time step restriction appears as the Courants(CFL) number criteria for the flow calculations. This difficulty can be overcome by introducing Runge-Kutta [1] time marching

schemes which allow larger CFL numbers. For unsteady flow problems, it would be appropriate to use higher order Runge-Kutta schemes which are stable for CFL number $C \leq 2\sqrt{2}$ if the dissipative terms are small enough.

The Rational Runge-Kutta scheme(RRK) for Euler as well as Navier-stokes solver has been developed by Morinishi and Satofuka [10]. For the flow domain, the accuracy of the solution can be improved by increasing the number of grid points. Increasing the number, on the other hand, may reduce the efficiency of the solution considerably. Since a smaller time step requires a greater number of iterative steps in addition to increase in operations counts per step. This difficulty has been overcome by Morinishi and Satofuka [10] by introducing RRK time stepping.

In this thesis work, the two stage RRK scheme has been used as a time-stepping scheme. With this scheme the solution is fully explicit as well as robust.

2.5.2 RRK Formulation

The Navier-Stokes equation discretized by Finite Volume principle, as shown in section 2.2, can be rewritten in simplest possible way as,

$$\frac{d\hat{q}}{dt} = \hat{Q}(\hat{q}) \quad (2.64)$$

For the integration of this ordinary differential equation, the two stage RRK scheme [10] is used. The two-stage RRK scheme can be written as,

$$\hat{g}_1 = \Delta t \hat{Q}(\hat{q}^n) \quad (2.65)$$

$$\hat{g}_2 = \Delta t \hat{Q}(\hat{q}^n + c_2 \hat{g}_1) \quad (2.66)$$

$$\hat{g}_3 = b_1 \hat{g}_1 + b_2 \hat{g}_2 \quad (2.67)$$

$$\hat{q}^{n+1} = \hat{q}^n + \frac{[2\hat{g}_1(\hat{g}_1, \hat{g}_3) - \hat{g}_3(\hat{g}_1, \hat{g}_1)]}{(\hat{g}_3, \hat{g}_3)} \quad (2.68)$$

where, the superscript n denotes the index of time steps and (\hat{g}_i, \hat{g}_j) denotes the scalar product of two vectors \hat{g}_i and \hat{g}_j . The coefficient b_1 , b_2 and c_2 must satisfy the relations:

$$b_1 - b_2 = 1 \text{ and } b_2 c_2 \leq -0.5$$

For an efficient second order scheme,

$$b_1=2, \quad b_2=-1 \text{ and } c_2=0.5 \text{ which have been used here.}$$

The time step Δt is determined locally so that the local CFL number becomes constant for each grid point. The critical time step for each coordinate direction is obtained as follows:

$$(\Delta t_\xi)_{i,j} = \left[\frac{1.0}{|U| + c\sqrt{\xi_x^2 + \xi_y^2}} \right]_{i,j} \quad (2.69)$$

$$(\Delta t_\eta)_{i,j} = \left[\frac{1.0}{|V| + c\sqrt{\eta_x^2 + \eta_y^2}} \right]_{i,j} \quad (2.70)$$

The critical time steps are used in the estimation of dissipative fluxes. The net time step for each grid point is determined from these critical time steps as,

$$\Delta t_{i,j} = C_o \min(\Delta t_\xi, \Delta t_\eta)_{i,j} \quad (2.71)$$

Here C_o is the local Courant number and c is local speed of sound.

The viscous terms of the Navier-Stokes equation and the dissipative terms are calculated only in the first stage of RRK scheme and frozen in the second stage. The boundary conditions are called after every stage.

Finally, we could write the equation (2.68) in the form of,

$$\hat{q}_{i,j}^{n+1} = \hat{q}_{i,j}^n + \hat{r}_{i,j} \quad (2.72)$$

where,

$$\hat{r}_{i,j} = \frac{[2\hat{g}_1(\hat{g}_1, \hat{g}_3) - \hat{g}_3(\hat{g}_1, \hat{g}_1)]}{(\hat{g}_3, \hat{g}_3)} \quad (2.73)$$

This $\hat{r}_{i,j}$ term called as a residual term in two-stage RRK scheme. For the fast acceleration of the convergence, residual terms play major role, which could be handled by various accelerating techniques, which are not used in this thesis.

2.6 Boundary Conditions

Proper boundary conditions enforcement is the most important part in the computation. Particularly, for internal transonic flows, the boundary conditions are critical for the correct solution to be computed.

Here, in this section, for a given cascade domain, four boundary conditions have been discussed. As shown in section 3.3.2, Dirichlet boundary condition has been enforced for density and momentum quantities at inflow boundary. For our case of physical domain, the following are the broad categories of the boundary conditions as shown in Figure (2.5).

- (i) Periodic boundary conditions at AB, EF and similarly on CD and GH.
- (ii) Solid wall boundary conditions at BC and FG.
- (iii) Far-field boundary conditions
 - a. Inflow boundary conditions on AE.
 - b. Outflow boundary conditions on DH.

2.6.1 Periodic boundary condition

As shown in Figure (2.5), the boundaries AB, EF and CD, GH are the boundaries where periodic boundary condition is implemented. Since, these boundaries are in common free flow boundaries to the other rows of turbine blades, the flux exchange at the cell $j=1$ will be same as at dummy cell $j=m$, which will be the first cell in η direction for second row of the turbine blades.

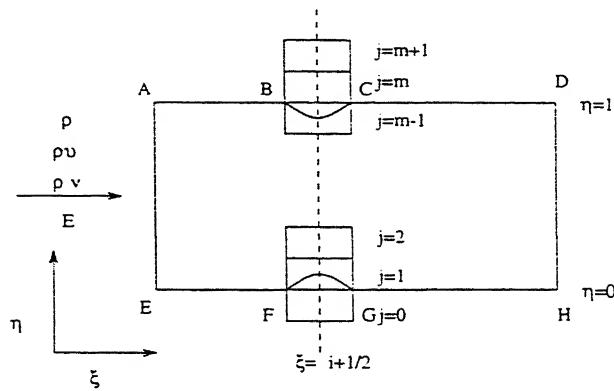


Figure 2.5: Periodic Boundary Zone

The fluxes for the other dummy cells also can be evaluated on the same basis. From the Figure(2.5),

$$\hat{q}_{j=0} = \hat{q}_{j=m-1}$$

and

$$\hat{q}_{j=m+1} = \hat{q}_{j=2}$$

for all i 's except in the solid boundary region.

2.6.2 Solid wall boundary condition

The Figure (2.5) shows, boundaries BC and FG as the solid wall boundaries in transformed plane. Solid boundaries are the actual blade zones. This zone is being formed with NACA0012 and DCA aerofoils for two different cases of study undertaken here.

For calculations of C_p values at upper and lower surface of aerofoils in Euler as well as in Navier-Stokes code, the boundary condition at solid wall is important.

On a solid wall, while computing inviscid flow in Euler code, it is necessary to satisfy the zero normal velocity boundary condition as shown by Ravichandran [11]. Similarly, for a viscous flow calculations both the tangential as well as

normal velocities should be forced to zero.

The contravariant component of tangential velocity near the wall is given as,

$$U = u\xi_x + v\xi_y \quad (2.74)$$

while normal velocity is given as,

$$V = u\eta_x + v\eta_y \quad (2.75)$$

Euler solution boundary conditions:

Since, present scheme is of cell center type, the solid wall forms a cell interface between $j = 1$ and $j = 0$ latter being the first of the two dummy cell rows introduced. A simple way to satisfy the zero normal velocity condition in inviscid flow on the cell interface is to use the reflection principle. This principle says that the normal velocity should be antisymmetric i.e.

$$V_{j=0} = -V_{j=1}$$

while other quantities (ρ, U, E) will remain symmetric for all dummy cells along the solid wall interface. The Figure (2.6) shows the reflection principle graphically.

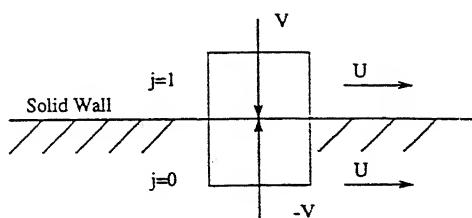


Figure 2.6: Solid Boundary condition for Euler calculations

With the above mentioned solid wall boundary condition, we get the values for $u_{j=0}$ and $v_{j=0}$ as,

$$u_{j=0} = \frac{U_{j=1}\eta_y|_0 + V_{j=1}\xi_y|_0}{J_0} \quad (2.76)$$

$$v_{j=0} = -\frac{U_{j=1}\eta_x|_0 + V_{j=1}\xi_x|_0}{J_0} \quad (2.77)$$

where, J is jacobian matrix and relation between ξ_x , η_x , ξ_y and η_y is,

$$(\xi_x)_{j=0} = (\xi_x)_{j=1}$$

$$(\xi_y)_{j=0} = (\xi_y)_{j=1}$$

$$(\eta_x)_{j=0} = (\eta_x)_{j=1}$$

$$(\eta_x)_{j=0} = (\eta_x)_{j=1}$$

and

$$J_{j=0} = J_{j=1}$$

Navier-Stokes solution boundary conditions:

Similarly, for Navier-Stokes solution both the contravariant velocity components should be antisymmetric across the solid surface for validation of reflection principle. As the flow is viscous in nature near the solid surface we treat tangential velocity component also to be antisymmetric. i.e.

$$U_{j=0} = -U_{j=1}$$

and

$$V_{j=0} = -V_{j=1}$$

While other quantities (ρ and E) are still retained symmetric across the solid surface boundary as shown in Figure (2.7). With the above mentioned boundary conditions

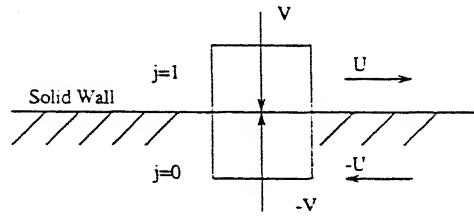


Figure 2.7: Solid Boundary condition for Navier-Stokes calculations

the equation (2.76) and (2.77) get modified for Navier-Stokes boundary conditions as,

$$u_{j=0} = \frac{-U_{j=1}\eta_y|_0 + V_{j=1}\xi_y|_0}{J_0} \quad (2.78)$$

$$v_{j=0} = -\frac{-U_{j=1}\eta_x|_0 + V_{j=1}\xi_x|_0}{J_0} \quad (2.79)$$

with all other constraints remaining same as of the Euler solver.

Requirement of the dummy cells is dependent on the order of the upwinding scheme used near the cells interface. For a third order MUSCL interpolation, we required two dummy cells while for a first order upwinding just one.

Temperature Variation across the Solid Surface:

In case of Navier-Stokes calculations temperature at solid boundaries is also the important boundary condition needed.

For our case of study, we are forcing an adiabatic condition on the surface. According to the adiabatic condition, the temperature gradient along the normal direction to the solid surface will be zero. As shown in Figure (2.8) the normal vector to the solid surface can calculated as,

$$\hat{n} = \hat{i} \sin\{\theta\} - \hat{j} \cos\{\theta\} \quad (2.80)$$

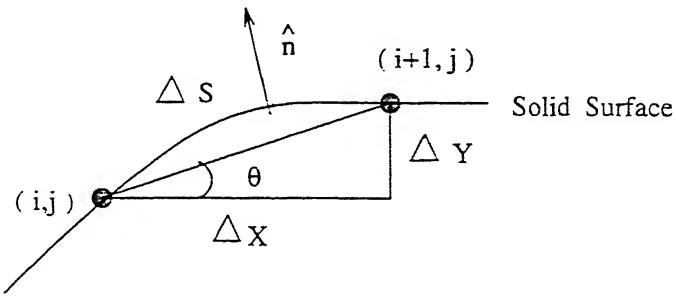


Figure 2.8: Normal Vector profile for the Solid Boundary Surface

where,

$$\sin\{\theta\} = \Delta y / \Delta s, \cos\{\theta\} = \Delta x / \Delta s$$

The temperature gradient for adiabatic condition can be calculated as,

$$\frac{\partial T}{\partial n} = \nabla T \cdot \hat{n} = 0 \quad (2.81)$$

since $\frac{\partial T}{\partial n} = 0$ is for adiabatic condition. which gives,

$$\sin\{\theta\} \frac{\partial T}{\partial x} - \cos\{\theta\} \frac{\partial T}{\partial y} = 0 \quad (2.82)$$

where,

$$\frac{\partial T}{\partial x} = \frac{\partial T}{\partial \xi} \xi_x + \frac{\partial T}{\partial \eta} \eta_x \quad (2.83)$$

$$\frac{\partial T}{\partial y} = \frac{\partial T}{\partial \xi} \xi_y + \frac{\partial T}{\partial \eta} \eta_y \quad (2.84)$$

Since we know the T_ξ at any point in solid boundary zone we can find out T_η from equations (2.82) to (2.84) as,

$$T_\eta = - \frac{T_\xi (\xi_x \sin\{\theta\} - \xi_y \cos\{\theta\})}{(\eta_x \sin\{\theta\} - \eta_y \cos\{\theta\})} \quad (2.85)$$

2.6.3 Far-field boundary conditions

Far-field boundary conditions for the present case are inflow and outflow boundary conditions. For inflow as well as at outflow boundary conditions, 1D Riemann Invariant [7] conditions are used.

Inflow Riemann Invariants

For the subsonic inlet condition, three Riemann Invariants are to be evaluated according to the free-stream value at upstream as they carry the information from outside into the flow domain while the fourth quantity is extrapolated from inside to the inflow boundary. In case of the transonic inlet condition the three flow variables ($\rho, \rho u, \rho v$) are evaluated from free-stream values and the fourth variable Energy (E) is calculated from three free-stream variables.

Inflow Riemann Invariants are calculated as [7],

$$R_1 = \left(\frac{p}{\rho}\right)_\infty \quad (2.86)$$

$$R_2 = q_{t\infty} \quad (2.87)$$

$$R_3 = \left(q_n + \frac{2c}{\gamma - 1}\right)_\infty \quad (2.88)$$

$$R_4 = \left(q_n - \frac{2c}{\gamma - 1}\right)_e \quad (2.89)$$

where subscript ∞ refers to as free-stream value, e is internal flow domain value, t is the tangential flow to the inflow boundary and n is the normal direction of the flow to the inflow boundary.

With the equation (2.88) and equation (2.89) we get the local normal velocity and speed of sound at inflow boundary as,

$$q_n = \frac{1}{2}(R_3 + R_4) \quad (2.90)$$

$$c = \frac{\gamma - 1}{4}(R_3 - R_4) \quad (2.91)$$

With these quantities we can obtain Energy at the inflow boundary.

Outflow Riemann Invariants

On a similar basis, we can calculate the outflow Riemann Invariants, where we have pressure as a fixed quantity. With fixed pressure at outlet, we can easily find out the Energy at outflow while other three quantities have to be extrapolated from inside the domain.

Outflow Riemann Invariants could be shown as.

$$p = p_{specified} \quad (2.92)$$

$$R_1 = \left(\frac{p}{\rho^\gamma}\right)_e \quad (2.93)$$

$$R_2 = q_{te} \quad (2.94)$$

$$R_3 = \left(q_n + \frac{2c}{\gamma - 1}\right)_e \quad (2.95)$$

These Riemann Invariants give the correct interpretation of the flow at outflow boundary.

2.7 Spectral Analysis of FVS Scheme

2.7.1 Introduction

It is well known for incompressible flows [14], any discretization scheme other than the one using spectral method introduces phase error and numerical dissipation. For high Reynolds number even spectral methods require numerical dissipation. While numerical dissipation is necessary for computing flows at high Reynolds number, it is

the phase error which determines the efficacy of the numerical method. The source of such errors are due to the treatment of the nonlinear convection terms in the conservation equations. We are not aware of similar analysis for compressible flows. In this section we extend the ideas in Sengupta & Sengupta [14], for the spectral analysis of the flux vector splitting schemes that are used in finite volume methods for the solution of compressible flows. Although the analysis is for the convection terms alone, the results would be valid for full Navier-Stokes solution. The analysis assumes greater significance since we use a higher order interpolation for the split fluxes.

2.7.2 Spectral Analysis

We analyse the schemes with respect to equation(2.2) given before. Here the inviscid fluxes are written in terms of the flux Jacobians A and B.

$$\frac{\partial \hat{F}_c}{\partial \xi} = A \frac{\partial \hat{q}}{\partial \xi} \quad (2.96)$$

$$\frac{\partial \hat{G}_c}{\partial \eta} = B \frac{\partial \hat{q}}{\partial \eta} \quad (2.97)$$

The unknowns \hat{q} are written using Fourier Spectral representation as given by,

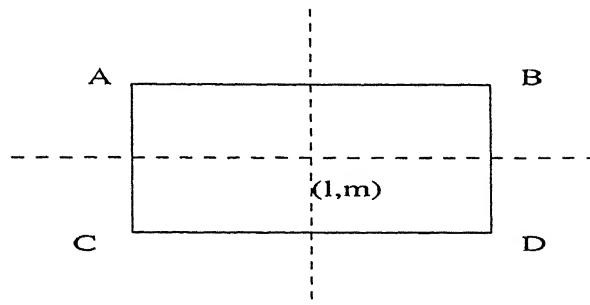
$$\hat{q}(\xi_l, \eta) = \int \hat{q}_1(k, \eta) e^{i \kappa \xi_l} dk \quad (2.98)$$

where $\xi_l = (l - 1) \Delta \xi$

Here we will only show the analysis with respect to one variable ξ , similar ideas can be used for the other direction as well.

After applying the finite volume technique, we actually solve the equation(2.4), where one of the fluxes are written as,

$$\hat{F}_c = \hat{A} \hat{q} \quad (2.99)$$

Figure 2.9: $(l, m)^{th}$ cell

For a given $(l, m)^{th}$ cell as shown below, let us illustrate the procedure when the above flux given by equation(2.99) is represented by,

$$\hat{A}\hat{q} = \hat{A}^+ \hat{q}^+ + \hat{A}^- \hat{q}^- \quad (2.100)$$

for the side AC.

In equation(2.100) the left hand side is evaluated as,

$$\hat{A}\hat{q} = \hat{A} \int \hat{q}_1 e^{i\kappa(l-3/2)\Delta\xi} dk \quad (2.101)$$

On the right hand side, the flux Jacobians are given by,

$$\hat{A} = \hat{A}^+ + \hat{A}^-, \quad \hat{B} = \hat{B}^+ + \hat{B}^- \quad (2.102)$$

with

$$\hat{A}^+ = R \Lambda_A^+ R^{-1}, \quad \hat{A}^- = R \Lambda_A^- R^{-1} \quad (2.103)$$

The various quantities are already defined in equation(2.11) to (2.16).

Similarly the unknown \hat{q} are split according to

$$\hat{q} = \hat{q}^+ + \hat{q}^- \quad (2.104)$$

with the split unknowns given as in equation(2.42) and (2.43).

Therefore the equation(2.100) can be rewritten as,

$$\begin{aligned}\hat{A}\hat{q}_{l-1/2,m} &= (R\Lambda_A^+ R^{-1})[\hat{q}_{l-1,m} + \frac{\epsilon(1-\kappa)}{4}\{\hat{q}_{l-1,m} - \hat{q}_{l-2,m}\} + \frac{\epsilon(1+\kappa)}{4}\{\hat{q}_{l,m} - \hat{q}_{l-1,m}\}] \\ &\quad + (R\Lambda_A^- R^{-1})[\hat{q}_{l,m} - \frac{\epsilon(1+\kappa)}{4}\{\hat{q}_{l,m} - \hat{q}_{l-1,m}\} - \frac{\epsilon(1-\kappa)}{4}\{\hat{q}_{l+1,m} - \hat{q}_{l,m}\}]\end{aligned}\quad (2.105)$$

where we have used the general representation for \hat{q}^+ and \hat{q}^- as given in Hirsch [9]

Since $A = R\Lambda_A R^{-1}$, the above equation can be also written as,

$$\begin{aligned}\frac{\text{ActualPhase}}{\text{EquivalentFVPhase}} &= \\ \frac{\Lambda_A^+}{\Lambda_A^- + \Lambda_A^+} &[e^{\frac{-i\kappa\Delta\xi}{2}} + \frac{\epsilon(1-\kappa)}{4}\{e^{\frac{-i\kappa\Delta\xi}{2}} - e^{\frac{-3i\kappa\Delta\xi}{2}}\} + \frac{\epsilon(1+\kappa)}{4}\{e^{\frac{i\kappa\Delta\xi}{2}} - e^{\frac{-i\kappa\Delta\xi}{2}}\}] \\ + \frac{\Lambda_A^-}{\Lambda_A^- + \Lambda_A^+} &[e^{\frac{i\kappa\Delta\xi}{2}} - \frac{\epsilon(1+\kappa)}{4}\{e^{\frac{i\kappa\Delta\xi}{2}} - e^{\frac{-i\kappa\Delta\xi}{2}}\} - \frac{\epsilon(1-\kappa)}{4}\{e^{\frac{3i\kappa\Delta\xi}{2}} - e^{\frac{i\kappa\Delta\xi}{2}}\}]\end{aligned}\quad (2.106)$$

Ideally the left hand side should be equal to unity. The used splitting shows that the right hand side is a combination of a real and an imaginary quantity. The departure of the real part from unity represents the phase error or wave number distortion. Physically such distortion gives rise to numerical dispersion. The imaginary part contributes to the added numerical dissipation, which is absolutely necessary for numerical stability.

In equation(2.106), various schemes can be simultaneously analysed depending on the value of ϵ and κ , which are discussed in details in section (2.5.2). For example, for the MUSCL interpolation, $\epsilon = 1$ and $\kappa = 1/3$ as shown in equation (2.50).

Furthermore in equation(2.106) the factor $\frac{\Lambda_A^+}{\Lambda_A^+ + \Lambda_A^-}$ is always positive if Λ is positive and greater than one. If this factor is denoted by $\alpha (> 1.0)$, the other factor in equation(2.106) will be given by $(1 - \alpha)$.

In Figures (2.10) and (2.11), the real and imaginary part of equation (2.106) have been shown for all α values i.e. irrespective of the value of the α , the real and imaginary solutions for different finite volume schemes remain identical.

Certain features emerge from this figure. Firstly the abscissa is plotted for $(\kappa\Delta\xi)$ up to a maximum value determined by the Nyquist criteria. Second feature is that the first order upwinding scheme($\epsilon = 0$) and the central differencing($\kappa = 1, \epsilon = 1$) show identical phase error, but the distortion is rather high. Notice that the first order upwinding is nothing but the second order central differencing scheme with implicit addition of numerical dissipation-as can be seen from the imaginary Figure (2.11). Thirdly some of the other schemes including the MUSCL interpolation scheme shows some overshoots in the plotted range. While this is an undesirable attribute of any scheme, for the used MUSCL scheme($\epsilon = 1, \kappa = 1/3$), this is minimum. The maximum overshoot is given in Table (2.1).

κ	ϵ	Scheme	Max. Overshoot
0	1	Linear Interpolation	10%
1	1	Central Differencing	no overshoot
1/2	1	QUICK	no overshoot
1/3	1	MUSCL interpolation	1%
-1	1	One sided linear interpolation	40%
Any	0	First order upwinding	no overshoot

Table 2.1: Phase error for different Interpolation schemes

It is evident that the one-sided linear interpolation($\epsilon = 1, \kappa = -1$), is practically useless as it has tremendously high built-in numerical dissipation. At the same time the numerical dissipation of this scheme is also the highest among the

plotted schemes at the high wave number range. Fourthly, when one compares the MUSCL interpolation scheme with the first order upwinding scheme, one notices that the latter damps across all wave numbers, while the MUSCL scheme only damps the higher wave numbers. High Reynolds number calculations requires the large eddies to be simulated without dispersion, while the numerical instability be removed by damping the high wave number components. Also notice that the linear interpolation($\kappa = 1, \epsilon = 1$), has phase error overshoot intermediate between MUSCL and one-sided linear interpolation. This case also has numerical dissipation that is in between the first order upwind scheme and MUSCL interpolation scheme. Finally, the QUICK scheme($\kappa = 1/2, \epsilon = 1$), seems to be optimal i.e. this scheme does not have any overshoot, at the same time has lower dissipation at lower wave number range. However if the added dissipation at the higher wave number range is adequate or not, to suppress nonlinear instabilities, can only be investigated with respect to a realistic simulation at high Reynolds numbers calibrated with respect to experimental results.

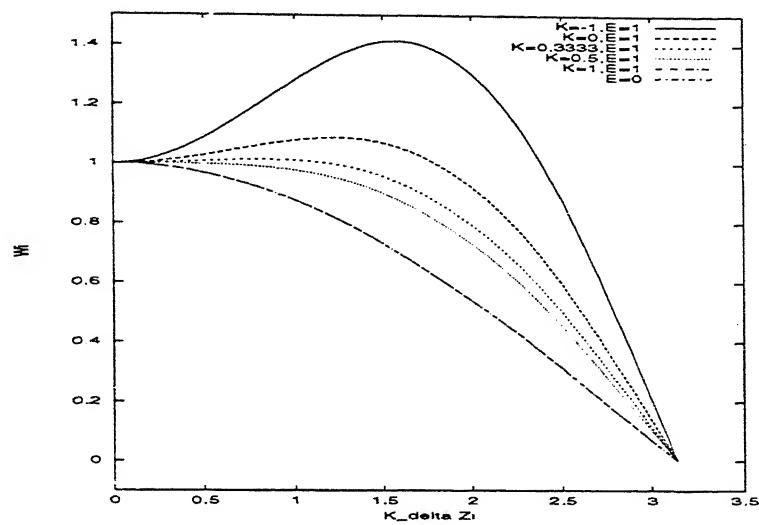


Figure 2.10: Phase error comparison for different schemes

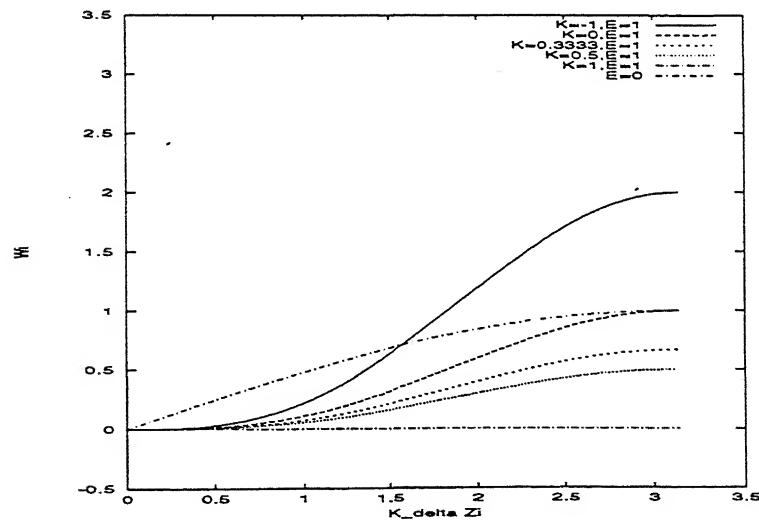


Figure 2.11: Numerical dissipation comparison for different schemes

Chapter 3

Grid Generation

3.1 Introduction

Grid generation is of importance to solve any flow problem for a given physical system involving complex geometries. Grids that are used can be classified into two major categories,

- (i) Structured grid,
- (ii) Unstructured grid.

Structured grids are basically used for solving flow equations based on Finite Difference or Finite Volume principles, while unstructured grids are best suited for Finite Element applications where there is no definite structure is maintained between nodal points. In this work we are concerned with the former.

Grid generation is a technique to establish the correspondence between point (x,y) in the Cartesian coordinate system and point (ξ,η) in the transformed computational domain. The problem of grid generation can be posed as a boundary value problem. The Figure (3.1) shows the boundary value problem has been based with

Dirichlet boundary conditions. In defining relationship between points in physical and computational domain i.e. $x=x(\xi, \eta)$, $y=y(\xi, \eta)$, it is necessary that there be a one-to-one correspondence. It would be unacceptable for a single point in physical domain to map into two or more points in computational domain and vice-versa. After establishing the mapping i.e. $x=x(\xi, \eta)$, $y=y(\xi, \eta)$, the requirement of one-to-one mapping can be determined by evaluating the determinant of transformation matrix, J , also known as *Jacobian*. For one-to-one mapping $| J |$ must be finite and nonzero at each and every point in the grid domain.

In this work, the flow equation based on the Finite Volume principle is solved using structured mesh with domain of (193×179) grid points for NACA0012 as well as DCA cascade domain as shown in Figure (3.3) to (3.5). The grid for the cascade problem has been generated by an algebraic grid generation technique and by elliptic grid generator.

The grid generated for NACA0012 aerofoil with (193×179) grid points by an algebraic technique is shown in Figure (3.3) in multiple domains. However such multiple domain grids will have discontinuous metrics. In this respect an elliptic grid generation method gives good grid smoothness and can be easily controlled for efficient grid spacing.

The grid generated for the cascade by an elliptic grid generation technique is shown in Figure (3.4). An elliptic grid has been found more suitable for solving Euler and Navier-Stokes equations.

3.2 Algebraic Grid generation

3.2.1 Introduction

The structured grid generation can be divided in to following categories,

- (i) Algebraic method
- (ii) Conformal mapping
- (iii) Method using PDE's

An algebraic method is efficient and can be applied to any 2D or 3D physical systems. This kind of mapping extrapolates the boundary data to generate interior grids. The major requirement is that the generated grids should be well conditioned i.e. smoothly varying, close to orthogonal. By means of some explicit grid control technique, we can have a desired grid spacing at critical points or zones in the domain. However, the slope discontinuities present at the boundaries tend to propagate to the interior. In such cases, the grid suffers from lack of smoothness. The distribution of the points along the boundary of the domain is handled effectively by defining normalized one-dimensional stretching functions along the boundary segments. One dimensional stretching function is described in next section.

3.2.2 One-Dimensional Stretching Functions

In an algebraic grid generation method, the transformation could be done analytically where the boundaries are regular. However, an interpolation technique has to be used for an irregular complex boundaries.

For the cascade geometry, we have treated whole domain as three blocks structure as shown in Figure (3.2).

- (a) An upstream zone(I),
- (b) Solid boundary zone(II),
- (c) A downstream zone(III).

We have generated the grid algebraically in these multiple blocks. As the algebraic grid generation process is very fast, the multiblock nature can be adapted for its suitability in solving the governing equation by domain decomposition, which

could be used in future for parallel computing.

Here, the grids are generated in domains using normalised one dimensional stretching functions, ratio analysis and bidimensional explicit interpolations. For the cascade, we consider the whole domain as union of three non-overlapping domains and grids are generated in each domain separately. Let us briefly discuss the stretching function that is used for distributing non-uniform points on the domain boundaries for the first block.

A normalised independent variable is defined as,

$$\xi^* = \frac{\xi - \xi_a}{\xi_b - \xi_a} \quad (3.1)$$

so that $0 \leq \xi^* \leq 1$ as $\xi_a \leq \xi \leq \xi_b$;

where a and b are the first and last points along the boundaries of the domain.

A stretching function is defined by Eiseman [1] as,

$$s = P\xi^* + (1 - P)\left[1 - \frac{\tanh[Q(1 - \xi^*)]}{\tanh Q}\right] \quad (3.2)$$

where P and Q as grid point control parameters. P effectively provides the slope of the distribution close to $\xi^*=0$; Q is called a damping factor and controls the variation from linear dependence. Once s is obtained, it is used to specify the distribution of x and y. For example, defining,

$$f(s) = \frac{x - x_a}{x_b - x_a}, \quad g(s) = \frac{y - y_a}{y_b - y_a}, \quad (3.3)$$

generates x(s) and y(s) directly, once the function f and g are prescribed. Here we have used ,

$$f(s) = g(s) = s \quad (3.4)$$

This procedure thus enables us in fixing the point distribution on the domain boundaries. The interior grid points are obtained from a double interpolation as given by Fletcher [1] which we are not discussing here in this thesis.

One can see the point distribution on domain boundaries formulated by stretching function as shown in Figure (3.3).

3.3 Elliptic Grid generation

• 3.3.1 Introduction

An elliptic grid generation is a general method of generating boundary fitted coordinate system with Dirichlet boundary conditions on all boundaries. In case of an elliptic PDE grid generation technique, one coordinate is specified to be constant on each of the boundaries and a monotonic variation of the other coordinate around each boundaries is specified. Poisson's equation in transformed plane of a linear elliptic system is used as a partial differential equation. Since only elliptic systems allow specification of a boundary conditions on the entirety of closed boundaries, it is used for any general closed boundary configurations.

3.3.2 Co-ordinate System Generation

Figure (3.1) shows the transformed plane of a grid for cascade domain. Transformed plane contains $\xi=\text{constant}$ and $\eta=\text{constant}$ lines. Hence our cascade domain is transformed from (x,y) to (ξ, η) coordinate system.

From the Figure (3.1), Γ_1 and Γ_2 are the solid boundaries. while Γ_3 and Γ_4 are Inflow and Outflow boundaries. $\Gamma_5, \Gamma_6, \Gamma_7$ and Γ_8 are the periodic boundaries. The governing equations are elliptic partial differential equations and more specifically the general Poisson's equation is given by,

$$\xi_{xx} + \xi_{yy} = P(\xi, \eta) \quad (3.5)$$

$$\eta_{xx} + \eta_{yy} = Q(\xi, \eta) \quad (3.6)$$

where, P and Q are the known functions used to control interior grid clustering.

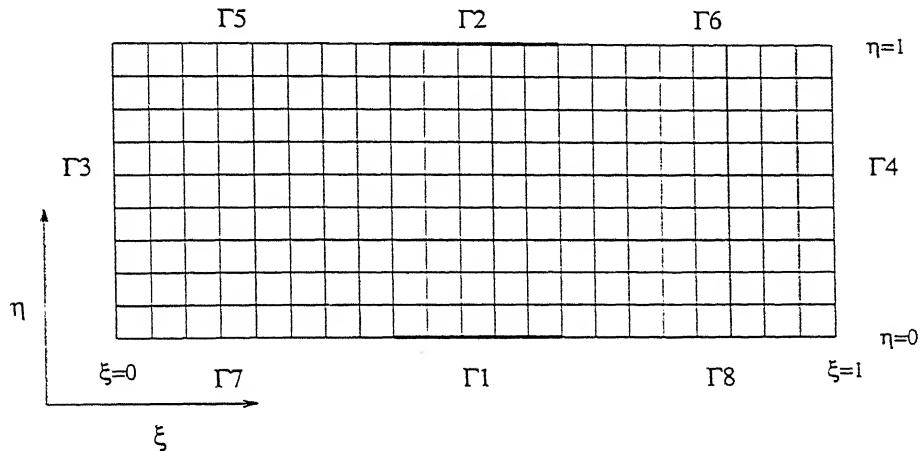


Figure 3.1: Transformed Plane of a domain

With the Dirichlet boundary conditions.

$$\begin{pmatrix} \xi \\ \eta \end{pmatrix} = \begin{pmatrix} \xi_1(x, y) \\ \eta_1 \end{pmatrix}, [x, y] \in \Gamma_1,$$

$$\begin{pmatrix} \xi \\ \eta \end{pmatrix} = \begin{pmatrix} \xi_2(x, y) \\ \eta_2 \end{pmatrix}, [x, y] \in \Gamma_2$$

and similarly for $\Gamma_3, \Gamma_4 \dots$ and Γ_8 , where $\eta_1, \eta_2 \dots$ and η_8 are different constants ($\eta_1 < \eta_2$) while $\xi_1, \xi_2 \dots$ and ξ_8 are specified monotonic functions on $\Gamma_1, \Gamma_2 \dots$ and Γ_8 respectively.

For the numerical computation in uniform rectangular transformed plane all the variables have to be changed accordingly. Thus the transformed plane equations are,

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} + J^2(Px_\xi + Qx_\eta) = 0 \quad (3.7)$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} + J^2(Py_\xi + Qy_\eta) = 0 \quad (3.8)$$

where,

$$\alpha = x_\eta^2 + y_\eta^2, \quad \gamma = x_\xi^2 + y_\xi^2,$$

$$\beta = x_\xi x_\eta + y_\xi y_\eta, \quad J = x_\xi y_\eta - y_\xi x_\eta.$$

With the transformed boundary conditions,

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f_1(\xi, \eta_1) \\ f_2(\xi, \eta_1) \end{pmatrix}, [\xi, \eta_1] \in \Gamma_1$$

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} g_1(\xi, \eta_2) \\ g_2(\xi, \eta_2) \end{pmatrix}, [\xi, \eta_2] \in \Gamma_2$$

and similarly for $\Gamma_3, \Gamma_4 \dots$ and Γ_8 , where the functions $f_1, f_2 \dots$ and $g_1, g_2 \dots$ are specified by the known shape of the contours $\Gamma_1, \Gamma_2 \dots \Gamma_8$ and specific distribution of ξ at these places.

The equation (3.7) and equation (3.8) give the quasi-linear elliptic system for physical coordinate functions, $x(\xi, \eta)$ and $y(\xi, \eta)$ in a transformed plane. With a recommendation by Thompson [1], Poisson's equation could be written as.

$$\begin{aligned} \xi_{xx} + \xi_{yy} &= - \sum_{i=1}^n a_i sgn(\xi - \xi_i) \exp(-c_i |\xi - \xi_i|) \\ &\quad - \sum_{j=1}^m b_j sgn(\xi - \xi_j) \exp(-d_j ((\xi - \xi_j)^2 + (\eta - \eta_j)^2)^{1/2}) \\ &= P(\xi, \eta) \end{aligned} \quad (3.9)$$

$$\eta_{xx} + \eta_{yy} = - \sum_{i=1}^n a_i sgn(\eta - \eta_i) \exp(-c_i |\eta - \eta_i|)$$

$$\begin{aligned}
 & - \sum_{j=1}^m b_j \operatorname{sgn}(\eta - \eta_j) \exp(-d_j((\xi - \xi_j)^2 + (\eta - \eta_j)^2)^{1/2}) \\
 & = Q(\xi, \eta)
 \end{aligned} \tag{3.10}$$

where the constants a_i , b_i , c_i and d_i are the grid lines controlling constants. In a given domain, close spacing between $\xi = \text{constant}$ lines for any $\xi = \xi_i$ line or lines in a domain can be achieved by selecting first term from equation (3.9) as.

$$P(\xi, \eta) = - \sum_{i=1}^n a_i \operatorname{sgn}(\xi - \xi_i) \exp(-c_i |\xi - \xi_i|) \tag{3.11}$$

The function sgn has the properties,

$\operatorname{sgn}(x) = 1$ if x is positive ,

$= 0$ if $x = 0$,

$= -1$ if x is negative.

Thus, the expression $\operatorname{sgn}(\xi - \xi_i)$ is $+1$ for $\xi > \xi_i$ and -1 for $\xi < \xi_i$. The constant a_i is maximum magnitude of P and it controls the extent by which ξ - grid lines shift towards $\xi = \xi_i$. The constant c_i is a decay constant of P which basically controls the specific zone or region for which the grid to be attracted. The constants a_i and c_i are not fixed for a physical system, but one has to get those values by trial and error until one gets the desired results.

If instead of attracting towards a complete $\xi = \text{constant}$ line. we would like to attract $\xi = \text{constant}$ line towards a given point (ξ_i, η_i) in domain, the control function P could be reduced from equation (3.9) to,

$$P(\xi, \eta) = - \sum_{j=1}^m b_j \operatorname{sgn}(\xi - \xi_j) \exp(-d_j((\xi - \xi_j)^2 + (\eta - \eta_j)^2)^{1/2}) \tag{3.12}$$

Similarly, choosing appropriate value of b_i and d_i , we have to work out by trial and error until we get satisfactory results. For the attraction to the line as well as to a point simultaneously, we have used full P and Q equations.

For attracting η -constant line towards a given η -constant line or a given point (ξ, η) in domain, the similar treatment is to be followed in the equation (3.10).

A precaution is to be followed in the selection of the constant values of a_i or b_i that if they are too large, the Extremum principle may get violated and the grid lines may cross each other. For adaptive grid generation, the control functions can be selected according to the error of gradient of predicted solution.

3.4 Results

The grid generated with an algebraic PDE technique has been brought to desired level of spacing accuracy at the aerofoil solid boundary zone for the η =constant lines with an elliptic PDE grid generation technique. By using equation (3.9), we get the spacing accuracy in η direction. As we have, aerofoils at $\eta=0$ and $\eta=1$ in transformed plane, we are attracting the η =constant lines towards η_i lines for normal spacing of the order of 10^{-3} near the aerofoil surface. Figures (3.3) and (3.5) show the algebraically generated grids with spacing in η direction close to aerofoil is 10^{-2} , which is controlled by elliptically generating the same grid for the spacing accuracy in η direction as 10^{-3} shown in Figures (3.4) and (3.6).

The constants of a_i 's and c_i 's are estimated by trial and error method which gives the desired level of spacing close to solid boundary of the order of 10^{-3} . For the present code we have used constants as tabulated in Table (3.1).

Grid	a_i	b_i	c_i	d_i
193×179	10.00	15.00	00.20	00.09
80×60	05.00	15.00	00.10	00.09

Table 3.1: Values of the constants for different grids

We have observed that for the cascade geometry, the constant values given above gives us the results shown in Figures (3.4) and (3.6). Any higher value of a_i or c_i will cause violation of Extremum principle giving inter-crossing grid line domain.

Due to sudden change in profile at leading and trailing edge, solution may diverge or there might be spurious oscillations of the solution. To avoid this, the smoothness of the grid lines near the leading edge and trailing edge is necessary. In the case of transonic or supersonic flows the shock wave generated at the leading edge while the expansion fan is found at trailing edge. To capture these shocks as well as expansion fans, we need to provide close grid spacing at leading and trailing edges as well as at $\xi_{lead}=\text{constant}$ and $\xi_{trail}=\text{constant}$ lines. In the code, we have forced the $\xi=\text{constant}$ lines to be attracted towards $\xi_{lead}=\text{constant}$ and $\xi_{trail}=\text{constant}$ lines by using equation (3.10). The point attraction is achieved by attracting $\eta=\text{constant}$ lines towards the $(\xi, \eta)_{lead}$ and $(\xi, \eta)_{trail}$.

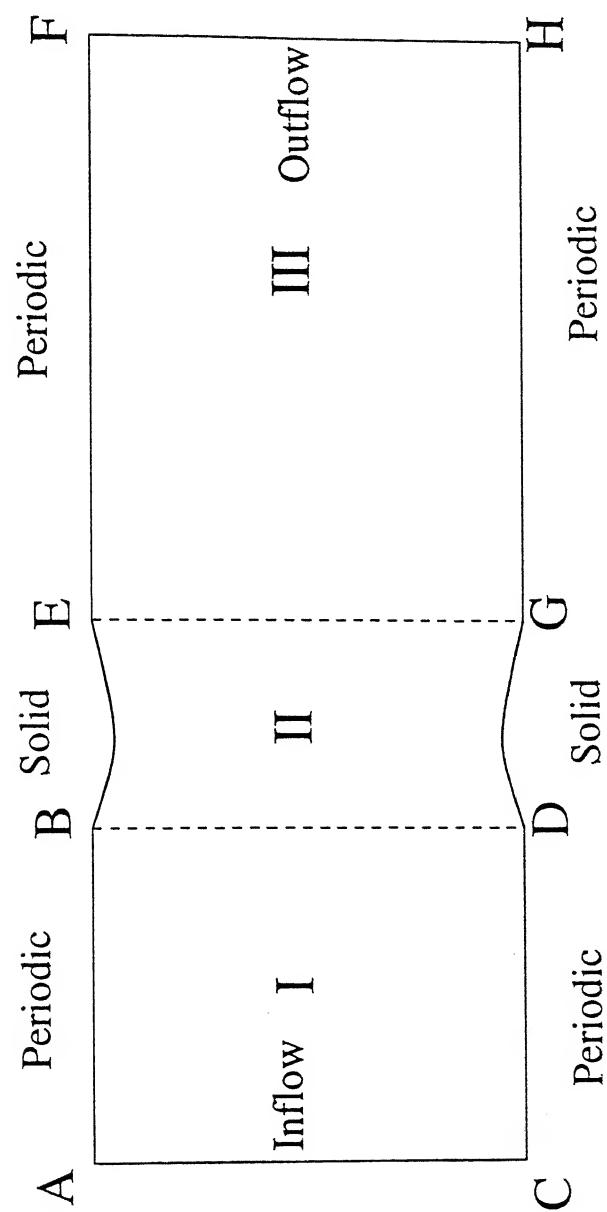


Figure 3.2: Straight Cascade Profile

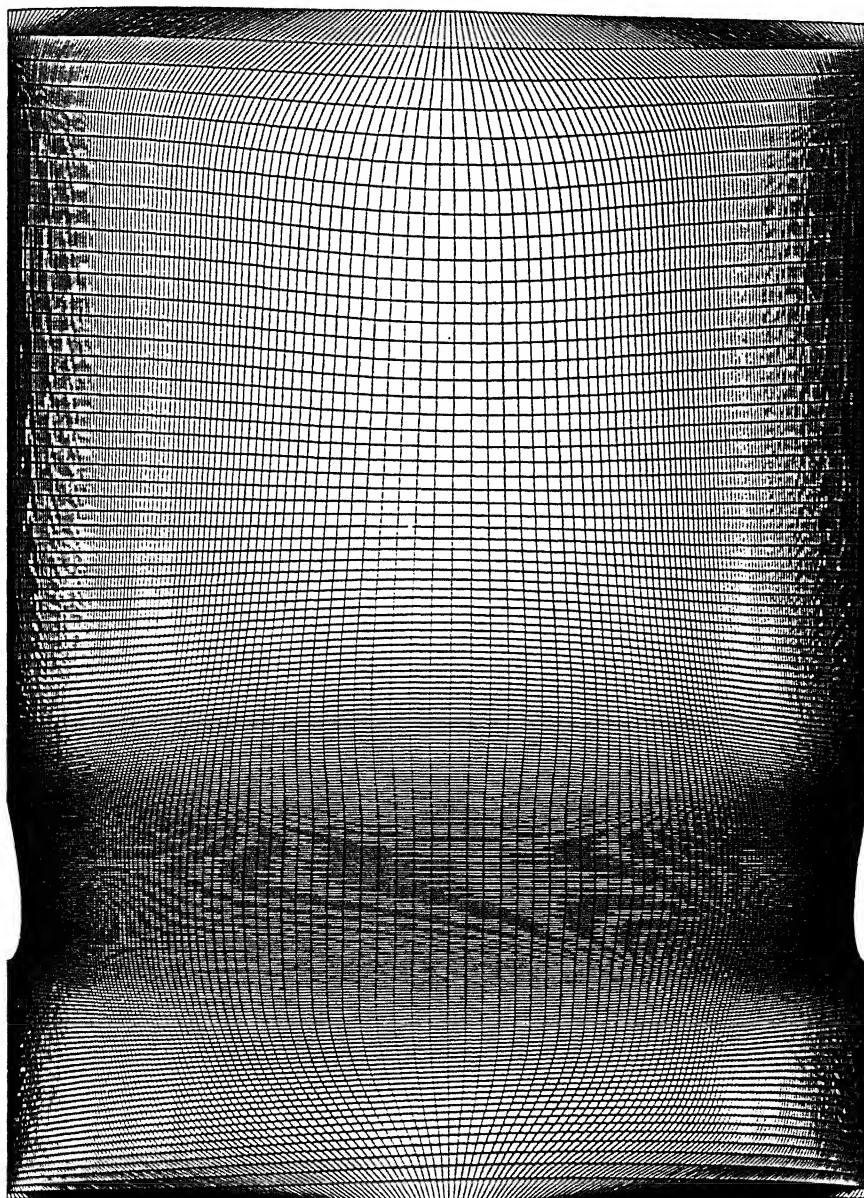


Figure 3.4: Elliptic grid for NACA0012 straight cascade $\alpha = 0^\circ$ and $\beta = 0^\circ$

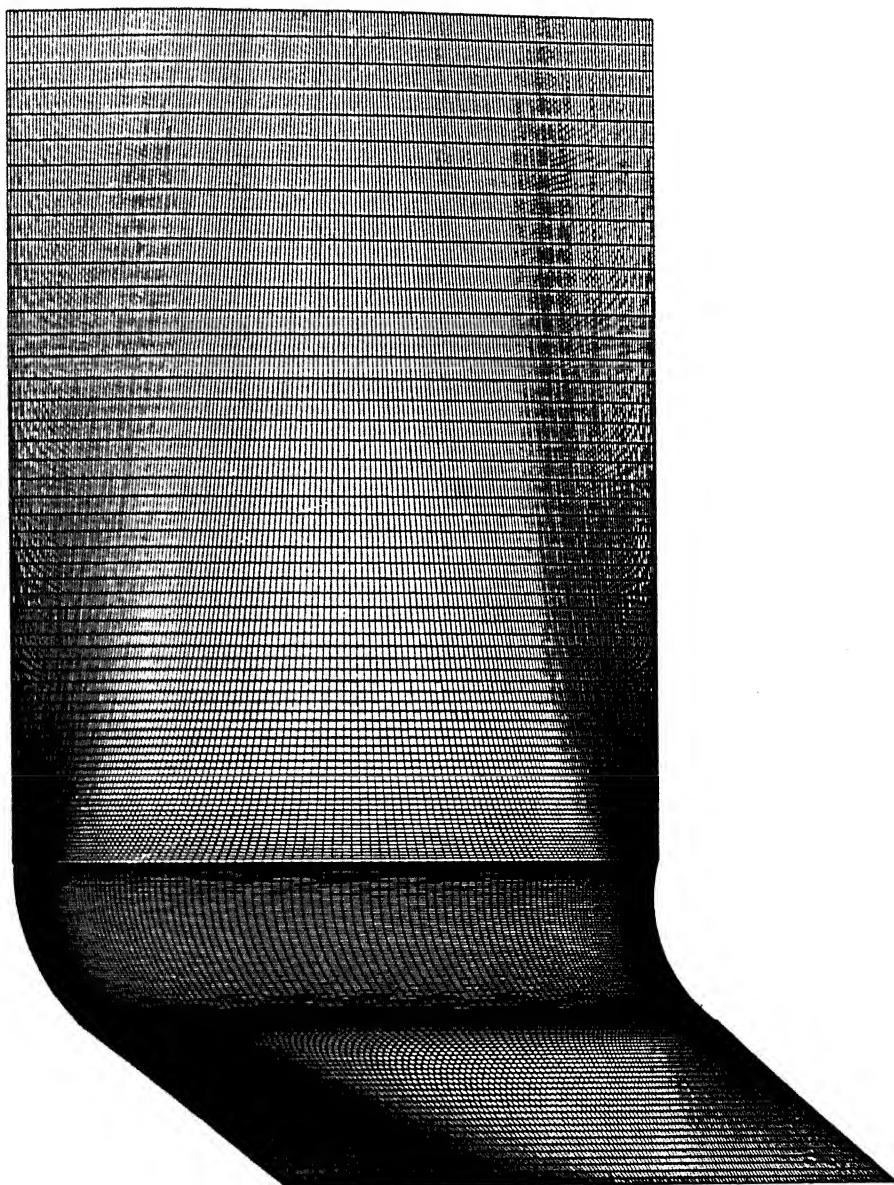


Figure 3.5: Algebraic grid for DCA cascade $\alpha = 139.5^0$ and $\beta = 0^0$

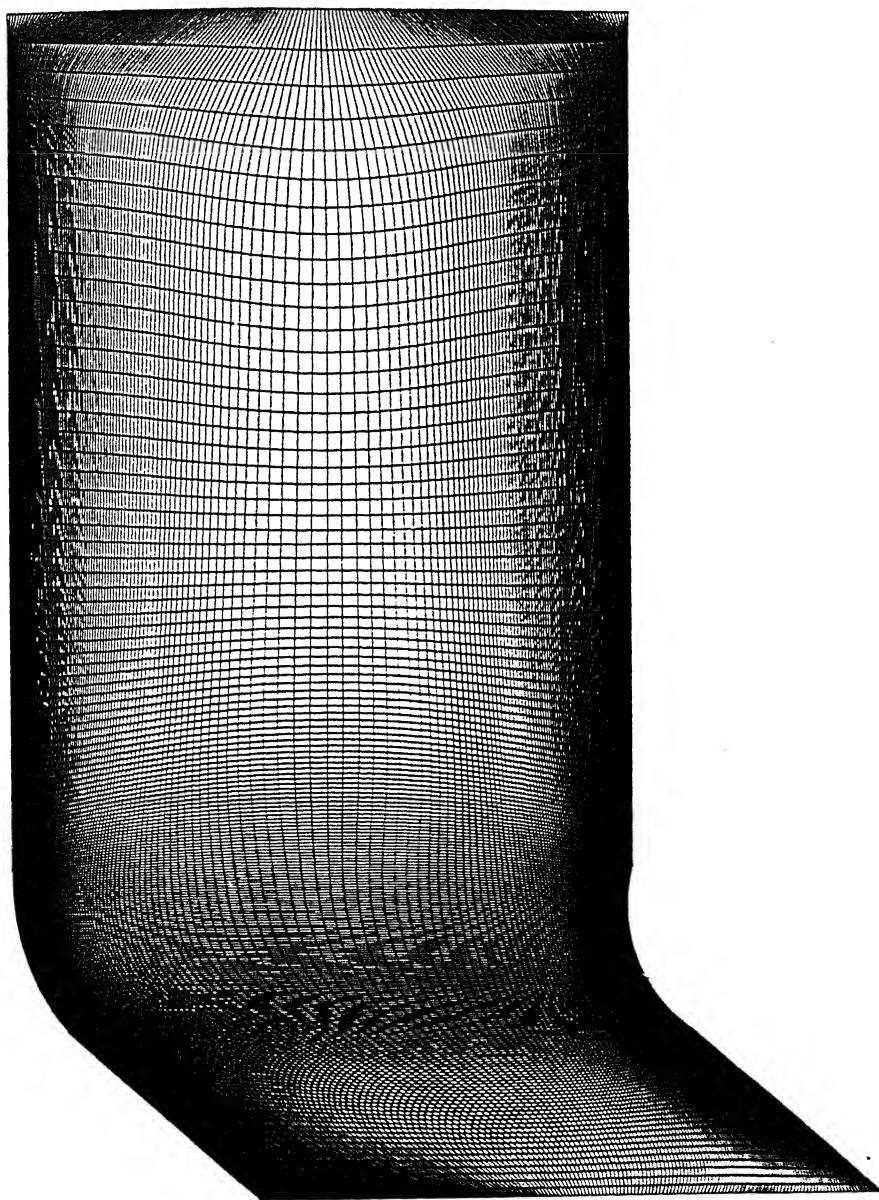


Figure 3.6: Elliptic grid for DCA cascade $\alpha = 139.5^\circ$ and $\beta = 0^\circ$

Chapter 4

Results and Discussion

A code for compressible flow is developed in FORTRAN to solve two dimensional unsteady viscous computations for flow past cascade. As a turbomachinery application, the code has been developed for internal flows which also can be used for any other physical domain for internal flows by choosing proper grid domain. The code is broadly divided into two parts. First, Euler code and second is the viscous code which adds the viscous terms along with a modifications in terms of boundary condition of the Euler code to give full Navier-Stokes solution. Since we have not used any local time stepping strategy, the results could be viewed as time accurate inviscid and viscous solutions respectively. However, we have used an initial condition which an uniform flow with $u = U_\infty$ and $v = V_\infty$ everywhere. In that context the intermediate results may not be very meaningful. Also such an initial condition will converge to actual steady state condition slowly. We recommend obtaining steady state Euler solution as a starting solution for Navier-Stokes computations.

4.1 Cases of Studies

We have carried out our computations for the following cases,

1. NACA0012 straight cascade:

This case was considered as a test case for which some computational results are already available in [7].

In this cascade domain, we have used parallel aerofoils so as to have a straight computational domain. i.e. ($\alpha = 0^0$ and $\beta = 0^0$), where α is the inflow angle and β is the outflow angle. The figure (3.2) shows the grid generated for straight cascade domain. All the computations for this configuration is carried out for a Mach number of 0.8. We have computed our solutions for two cases. First is for Euler computations with (80×60) grid domain and second is for Navier-Stokes computations for (193×179) grid domain. The grid spacing close to the solid wall for (80×60) domain is of the order of 10^{-2} , while for (193×179) domain it is of the order of 10^{-3} . For Euler computations, we have used CFL number as 10^{-3} , while Navier-Stokes computations are carried out for CFL number of 10^{-5} . For Euler computations, we have results for the inflow-outflow pressure ratio of 0.8. All Navier-Stokes computations are carried out for Reynolds number of 3×10^5 which is typical for compressor cascade operations.

2. DCA cascade:

This case of study is one of the cases documented in AGARD report [6].

For this case of study an experimental set-up is as follows:

Aerofoil profile	:	DCA
Number of blades (N)	:	14
Chord length (c)	:	80 mm
Aspect ratio (h/c)	:	3.75

Inflow angle (β_1)	:	139.5^0
Outflow angle (β_2)	:	90^0
Blade turning angle (β_s)	:	109.2^0

Navier-Stokes equation is being solved for Reynolds number 5×10^5 with a Mach number of 0.3. The grid has been generated for (193×179) domain with the solid wall spacing of 10^{-3} . The computed solution is compared with the experimental results shown by Fottener [6]. The computations are carried out for CFL number of 10^{-4} .

4.1.1 Euler Solution for NACA0012 Cascade

Euler computations are carried out to analyse the pressure variation along the aerofoil surfaces. Figures (4.1) to (4.4) give the pressure distribution calculated in terms of Coefficient of Pressure(C_p) along the upper and lower surfaces of the NACA0012 aerofoil. These figures show the pressure distribution for various different time step levels.

Figures (4.1) and (4.2) shows the C_p variations for a inflow-outflow pressure ratio of 0.8. We could see that the solution after $t = 1.0$ has started capturing the weak shocks around third quarter of the chord. These shocks are due to the transients at that location. After $t = 2.700$, we could see that the sharp decrease in C_p value from 0.6 to 0.2 around 0.65 of the cord length. It has been seen that the shocks are shifting from their previous locations as the iterations go on. The Figures at $t = 4.5$ to 5.0 give an idea about the shifting of this initial transient shock in downstream direction. From the Figure at $t = 10.0$, we could see that the solution approached steady state as the shock move away from the cascade. The C_p value variations from $t = 12.2$ to $t = 19.8$ show very little change, but as the solution

has not yet reached its convergence, we would see further changes in solution as iterations go on. It is to be emphasis that unlike the external flows, the internal flows approach their steady state value very slowly [8].

4.1.2 Navier-Stokes Solution for NACA0012 cascade

For a CFL number of 10^{-5} , the pressure variation along the solid surface is shown in figures (4.3) to (4.8). It can be seen from these figures that, as the marching time step is very small, it takes longer time to show any significant changes between successive frames.

From Figure (4.3), the results at $t = 0.01$ shows a primary oscillations around leading edge which are wiped out by $t = 0.03$ to produce smooth C_p value variation. From $t = 0.05$ onward we could see a dip near the leading edge which strengthens as iterations proceed. The comparison of figures at $t = 0.05$ and $t = 0.15$ for C_p shows significant change near the leading edge. The comparison of Figures between $t = 0.15$ and $t = 0.25$ shows that the dip has been shifted from 0.18 to 0.3 of the chord length, while the maximum C_p value is shifted from 0.98 to 0.95 of the chord length. We expect that the dip will continue to shift as well as steepen the C_p value at 0.7 of the chord.

4.1.3 Navier-Stokes Solution for DCA cascade

For DCA cascade, Figures (4.9) to (4.17) show the pressure variation in terms of C_p along the aerofoil surface for $M=0.3$, while Reynolds number is kept constant i.e. 10^5 .

From the Figures at $t = 0.1$ to $t = 0.5$, we could see that the initial transients in the solution show rapid change in C_p values. At $t = 0.2$, we could observe the

fluctuations at the leading edge for lower surface C_p value, which retain its location up to $t = 0.50$. The upper surface C_p , shows smooth variation from $t = 0.1$ to $t = 0.2$ which after $t = 0.24$, shows fluctuating up to $t = 0.50$.

After $t = 0.50$, we can see that both upper and lower surface C_p values have reached a maximum value at $x/c=0.9$. These maxima start shifting from trailing edge to leading edge as time progresses. The comparison of $t = 0.61$ and $t = 0.95$ shows that the maximum C_p values are shifted from 0.9 to 0.7 for upper surface and from 0.9 to 0.65 for lower surface. We could even observed that the maximum value of C_p comes down as iterations proceed. Figures for $t = 0.72$ to $t = 0.95$ show that some wiggles are propagating from leading edge to trailing edge, which are pushing upper surface C_p value to form a sharp peak as a weak shock. At $t = 1.34$, we could see the peak C_p value which has shifted to half chord, while C_p lower maximum has also shown similar shift to 0.3 of the chord.

From the Figures $t = 1.34$ to $t = 1.62$ show that the C_p lower values are smooth, while the shock on the upper surface moves toward leading edge. The comparison between $t = 1.66$ and $t = 1.98$ shows that the C_p upper value curve displays weakening of the shock while shifting towards the leading edge. The maximum C_p upper value is reducing as the iterations proceed. We could also see that the C_p lower value curve is also smooth with the minimum value shifted from 0.4 of the chord to 0.2 of the chord. After every 100 iterations we could see the comparable changes in the solution.

As this computations are not yet over, we could expect from the solution pattern that the C_p lower value will go further down to get minimum value at 0.7 of the chord, while C_p upper will show the maximum value at around 0.2 of the chord with the weak shock pattern.

4.2 Conclusions

1. Grids for the cases presented are generated by an algebraic as well as an elliptic grid generation technique. Although, an algebraic grid is much faster to generate, an elliptic grid gives smoother variations for J and hence a faster convergence. To capture accurate solutions we should have very fine grid spacing near the solid boundary. For our cases of study, we should use much more refined grid close to the solid surface in comparison with the present grid spacing.

2. We have used higher order interpolation upwinding scheme in calculating flux matrices. The spurious variations in the solution for higher Mach numbers could be minimised by using nonlinear Flux-Limiter(FL) approach [20]. This flux-limiter controls the Gibb's phenomenon generated for higher wave numbers(κ).

3. The computations are carried out for straight NACA0012 cascade as well as DCA cascade. From the results we could see that the straight cascade is more difficult to compute compared to DCA cascade, since the variations in the solution are rather slow. For the DCA cascade study case the results are compared with the experimental [6] case study. As the computations are still going on, we could expect from the results so far produced, the solutions should approach to the experimental results.

4.3 Future Scope of the Work

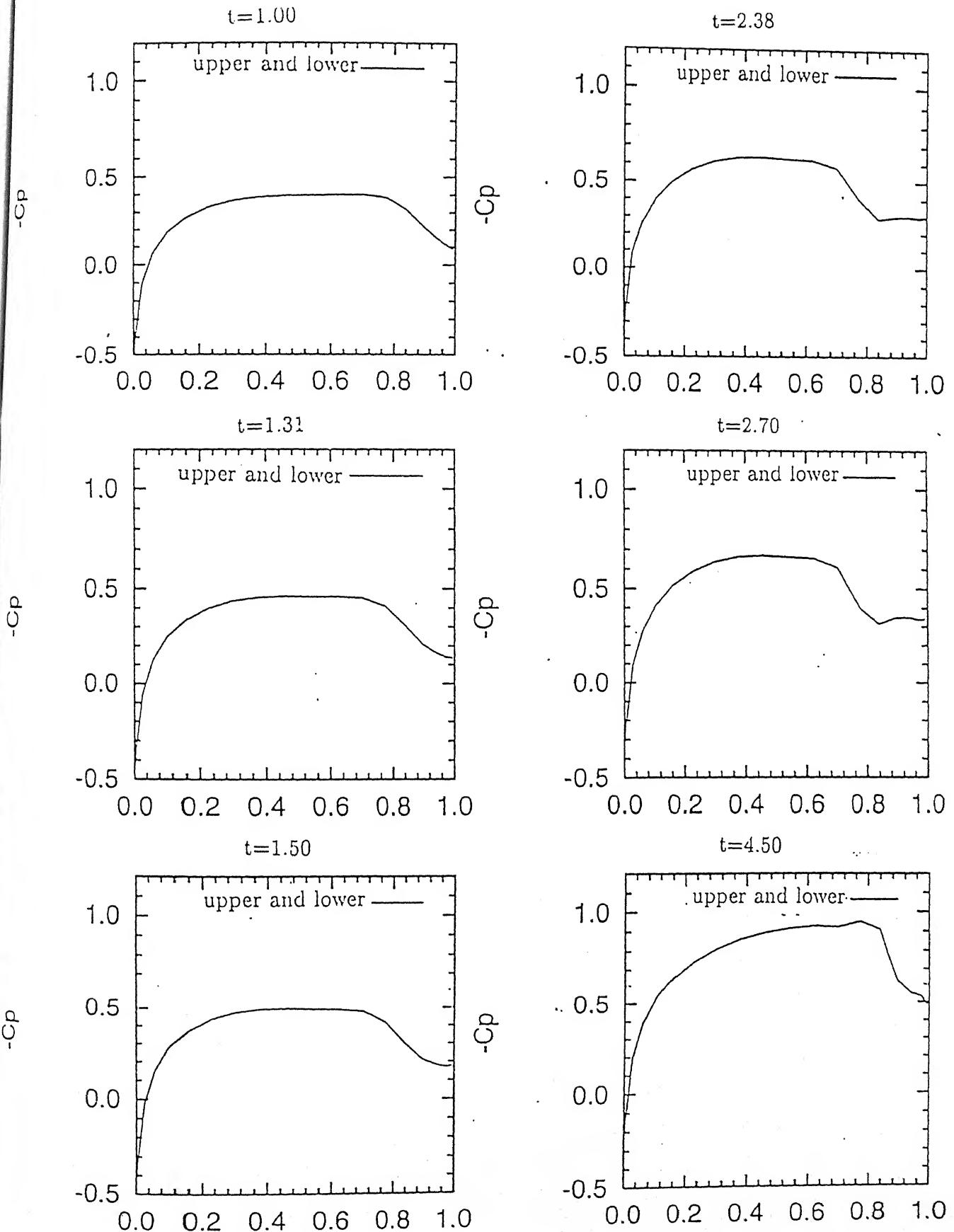
We have studied and analysed the results for straight as well as angular cascade domain with transonic as well as subsonic inflow Mach numbers.

In future, we would like to investigate solution for the high Mach number inflows at transonic or supersonic speeds. In the present work, our outflow boundary conditions are predefined, but in actual flows, the outflow velocity angles define the

boundary parameters at outflow. This feature can be incorporated in our code to achieve exact simulation of the turbine flow. We would like to study following aspects regarding the cascade flows in future:

1. Noise added to inflow to study effect of free-stream turbulence.
2. Take the outflow solution of the present set of computation as an inflow to another computation to investigate effect of gust.
3. Here periodicity is over row of cascades- it can be extended for multiple rows to study events like rotating stall.
4. Since we are obtaining time accurate solution, the present set of results can be used for studying aeroelastic and aeroacoustic aspect of cascade flows.

4.4 Figures



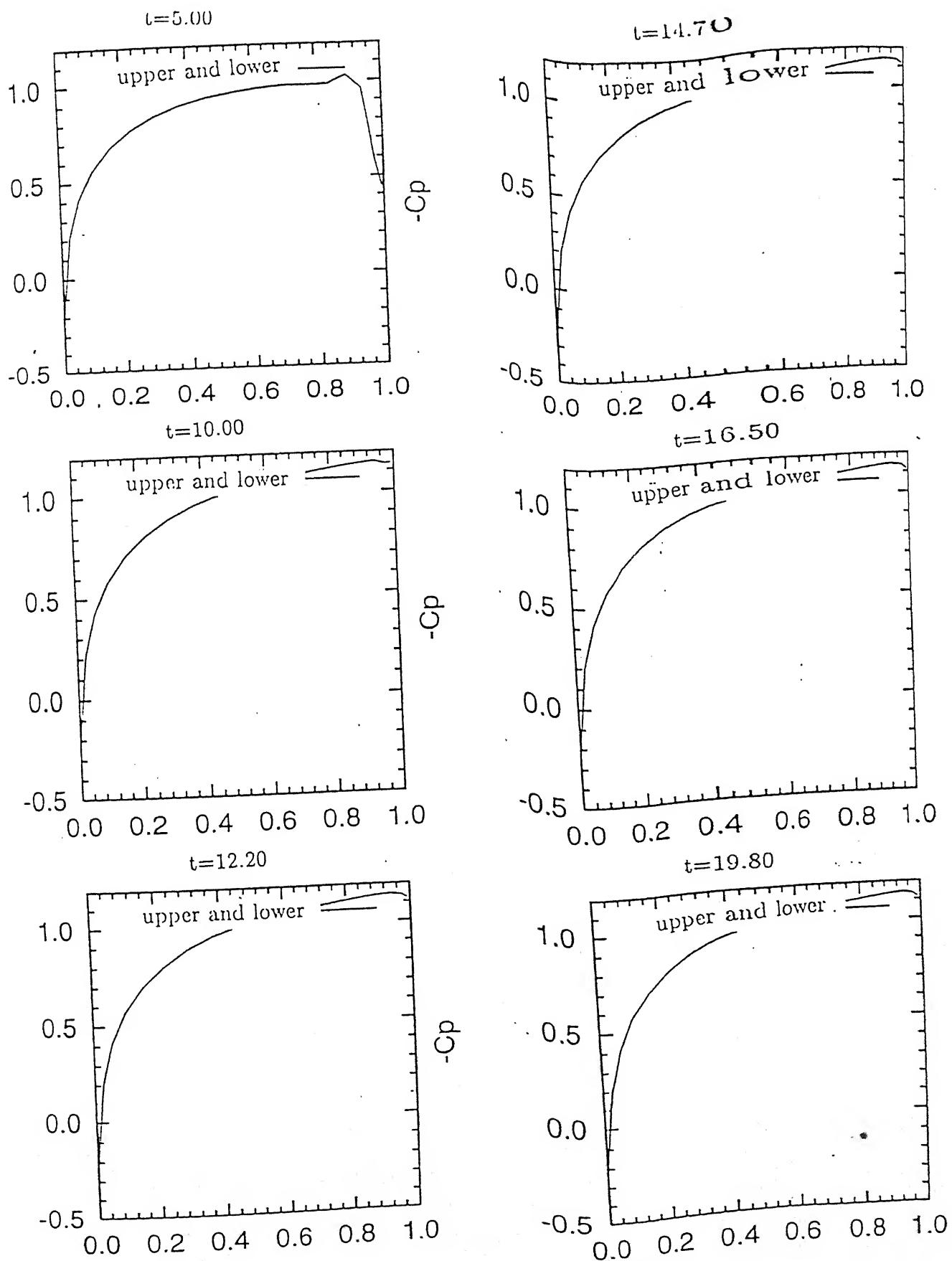


Figure 4.2: Pressure Variation for NACA0012 Euler Solution. Mach No. = 0.8,
 $CFL = 10^{-3}$

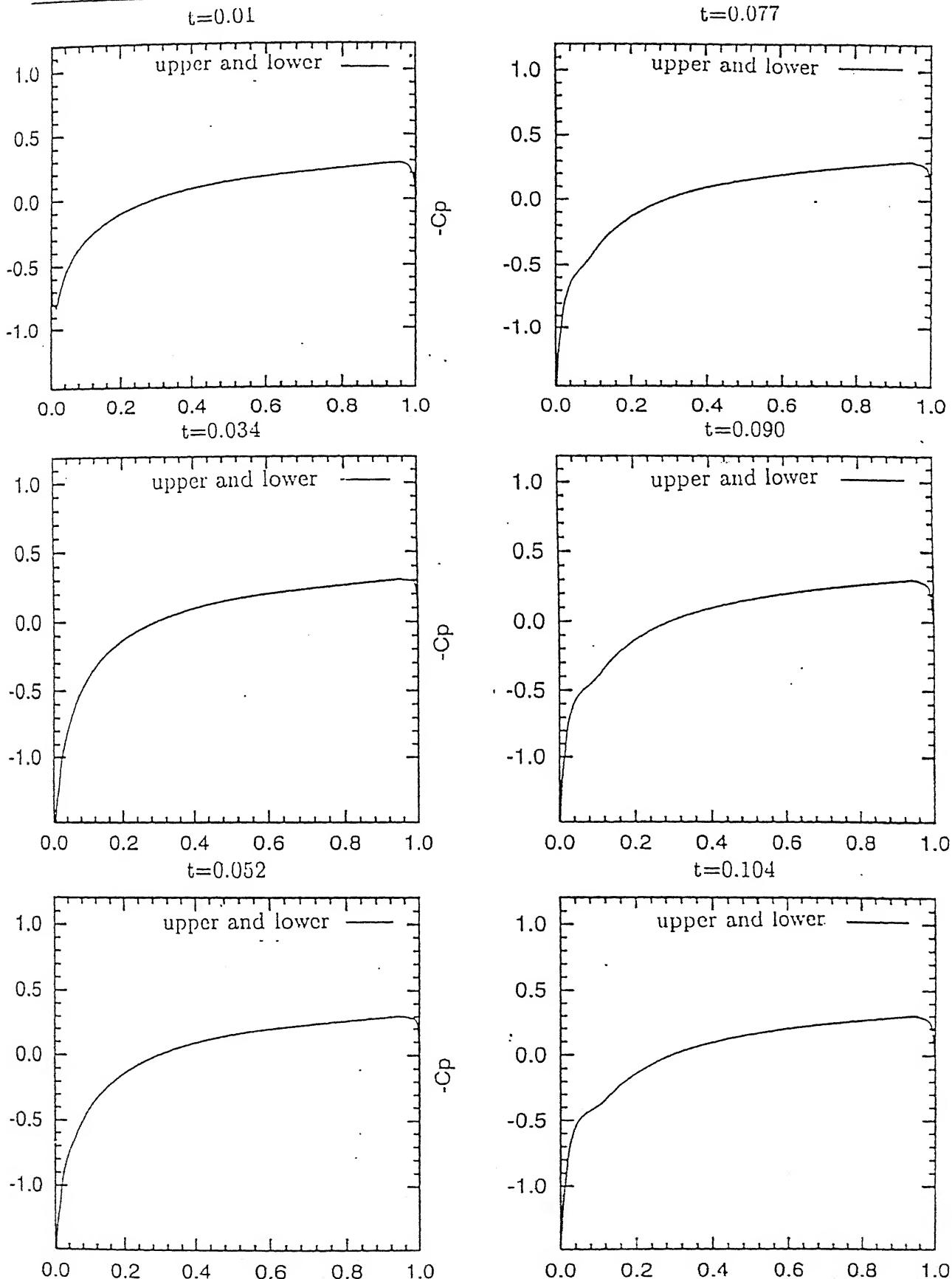


Figure 4.3: Pressure Variation for NACA0012 Navier-Stokes Solution *Contⁿ*. Mach No.= 0.8, CFL= 10^{-5}

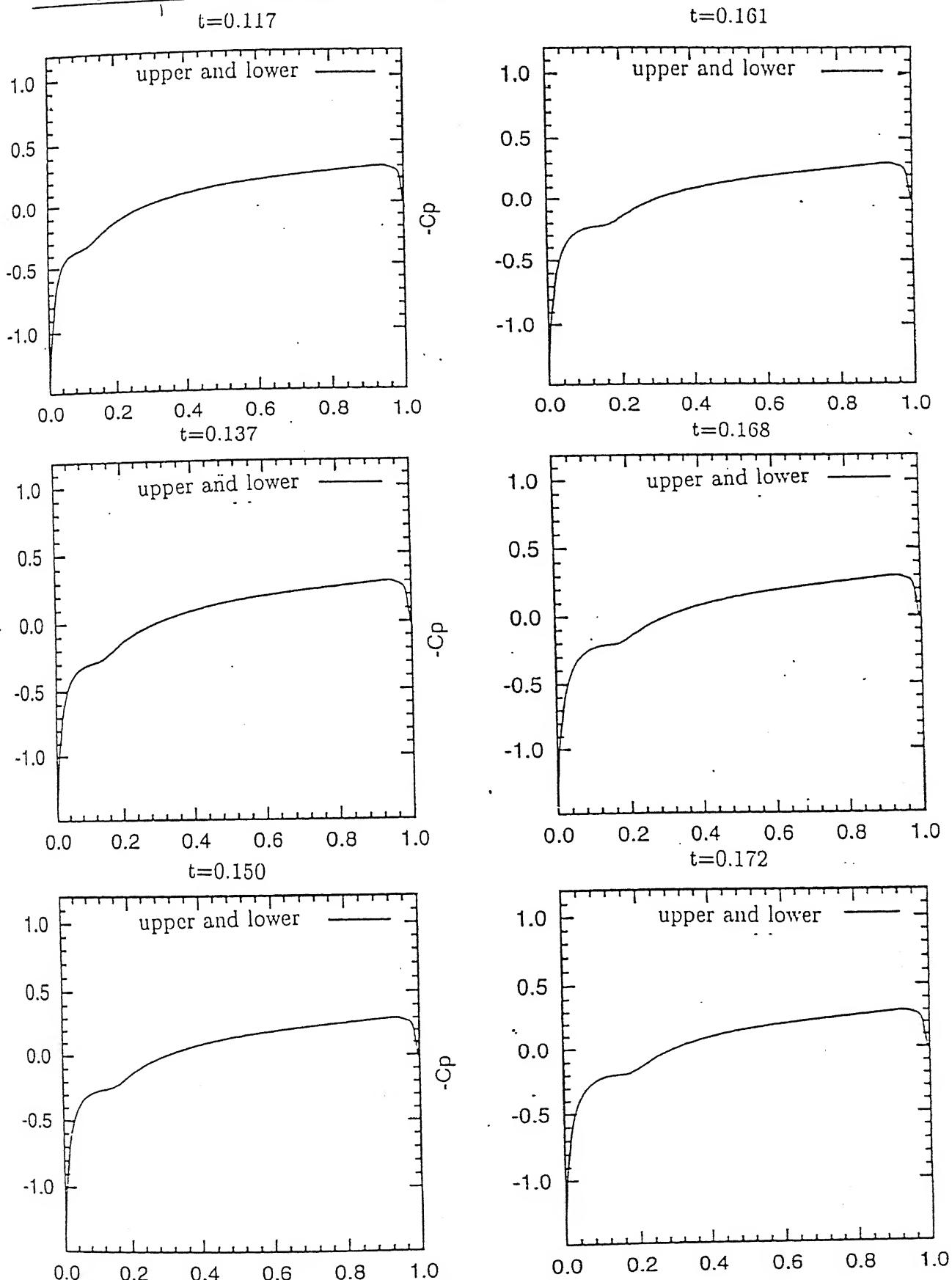


Figure 4.4: Pressure Variation for NACA0012 Navier-Stokes Solution *Contⁿ*. Mach No.= 0.8, CFL= 10^{-5}

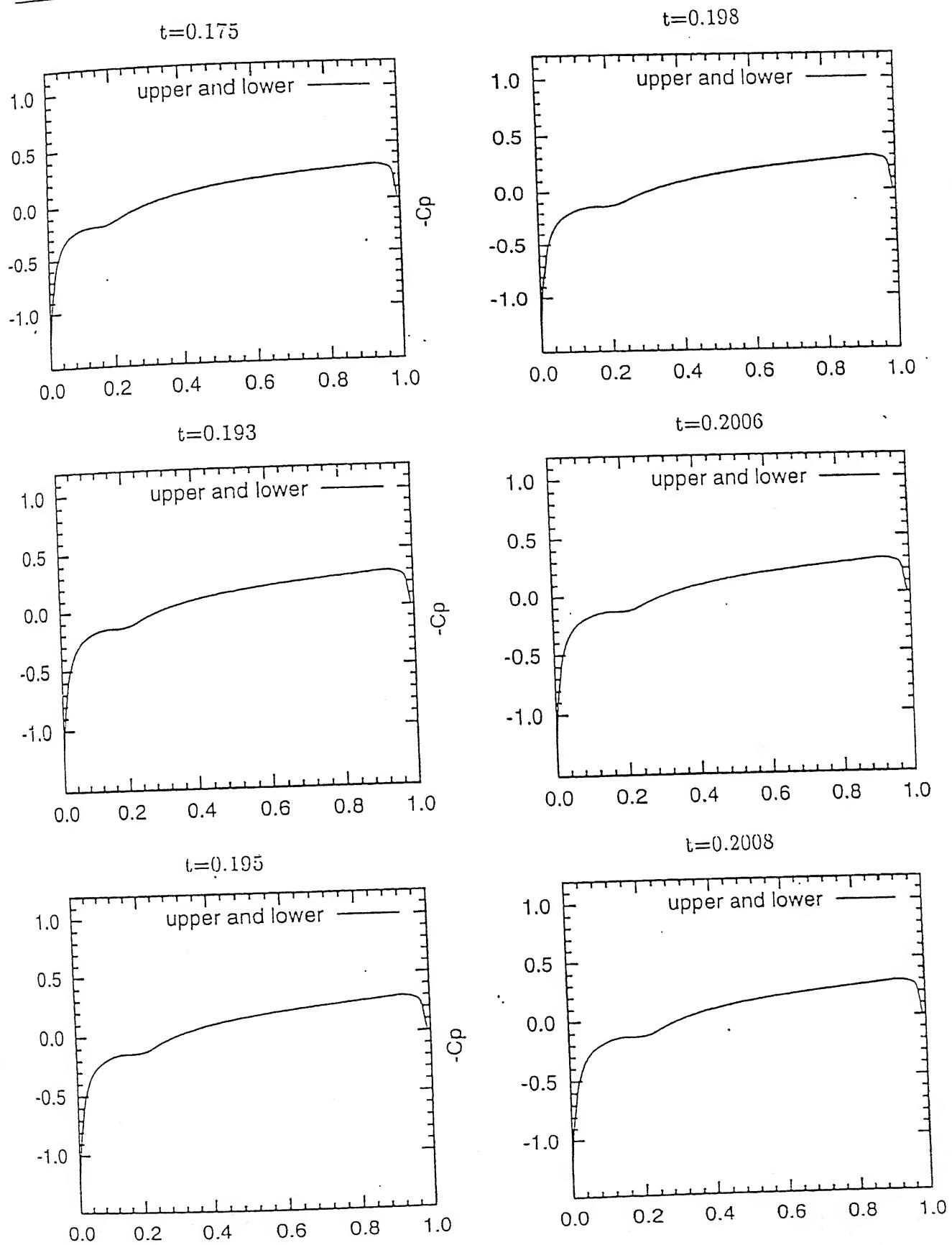


Figure 4.5: Pressure Variation for NACA0012 Navier-Stokes Solution *Contⁿ*. Mach No.= 0.8, CFL= 10^{-5}

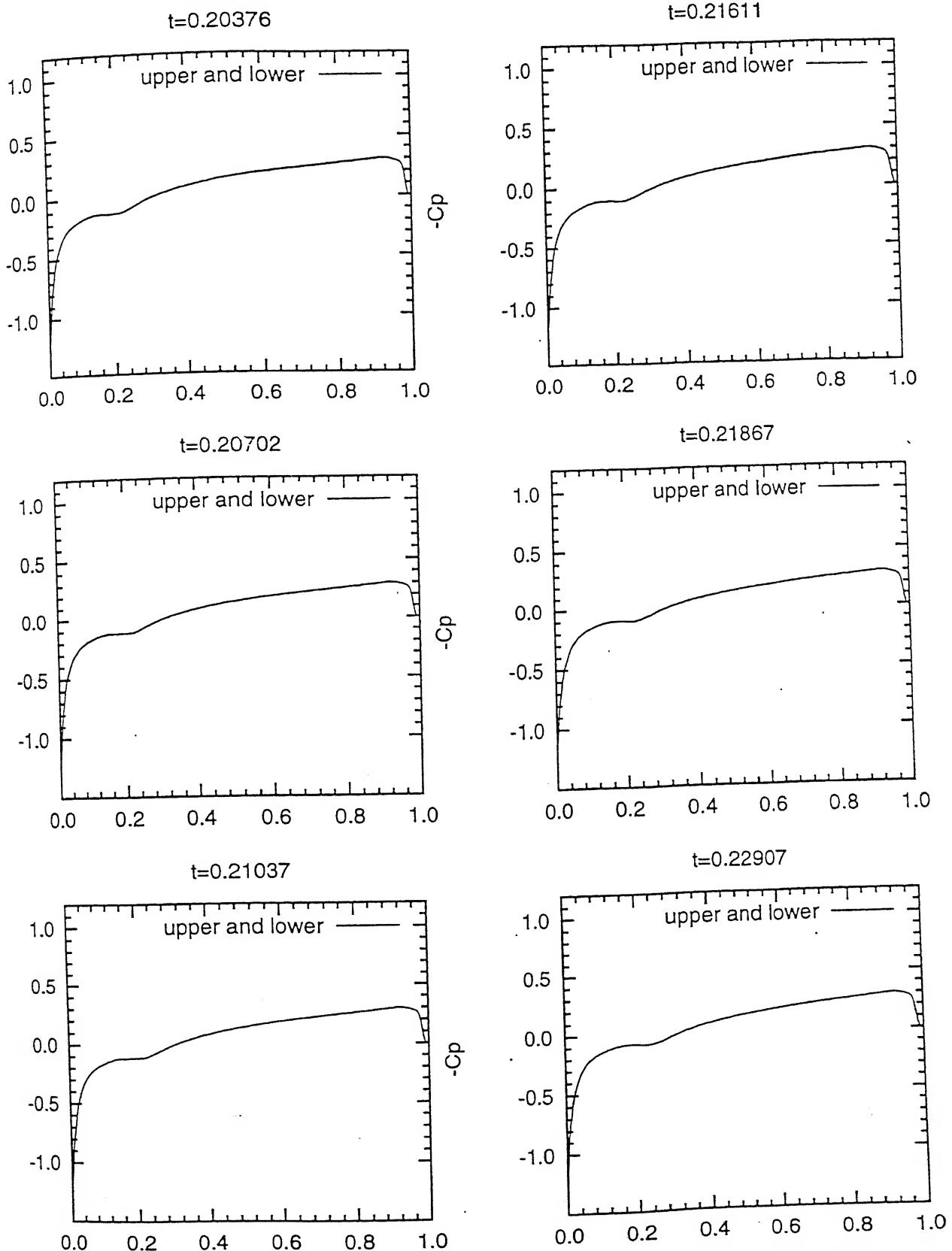


Figure 4.6: Pressure Variation for NACA0012 Navier-Stokes Solution *Contⁿ*. Mach No.= 0.8, CFL= 10^{-5}

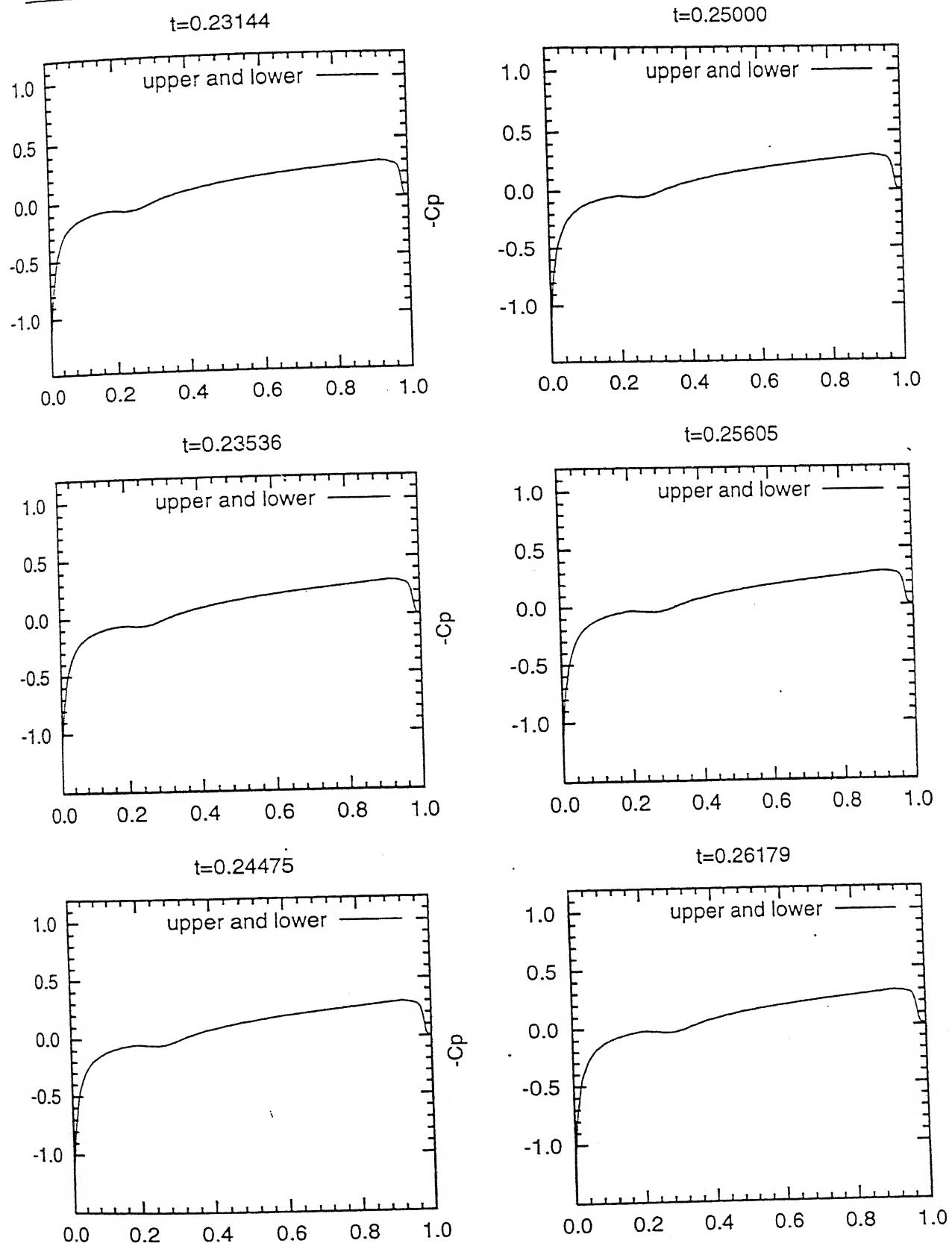


Figure 4.7: Pressure Variation for NACA0012 Navier-Stokes Solution *Contⁿ*. Mach No.= 0.8, CFL= 10^{-5}

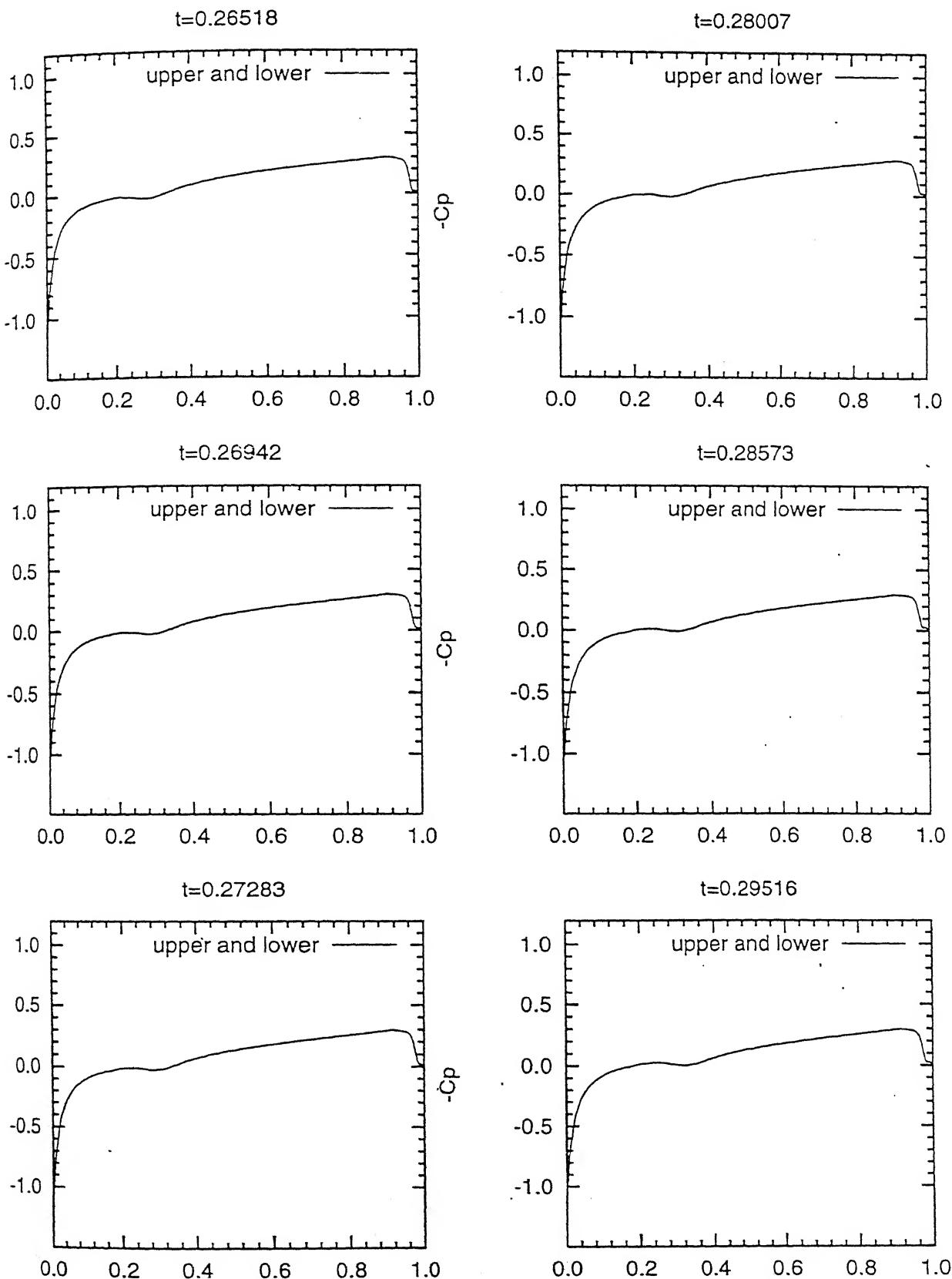


Figure 4.8: Pressure Variation for NACA0012 Navier-Stokes Solution *Contⁿ*. Mach

No. = 0.8, CFL = 10^{-5}

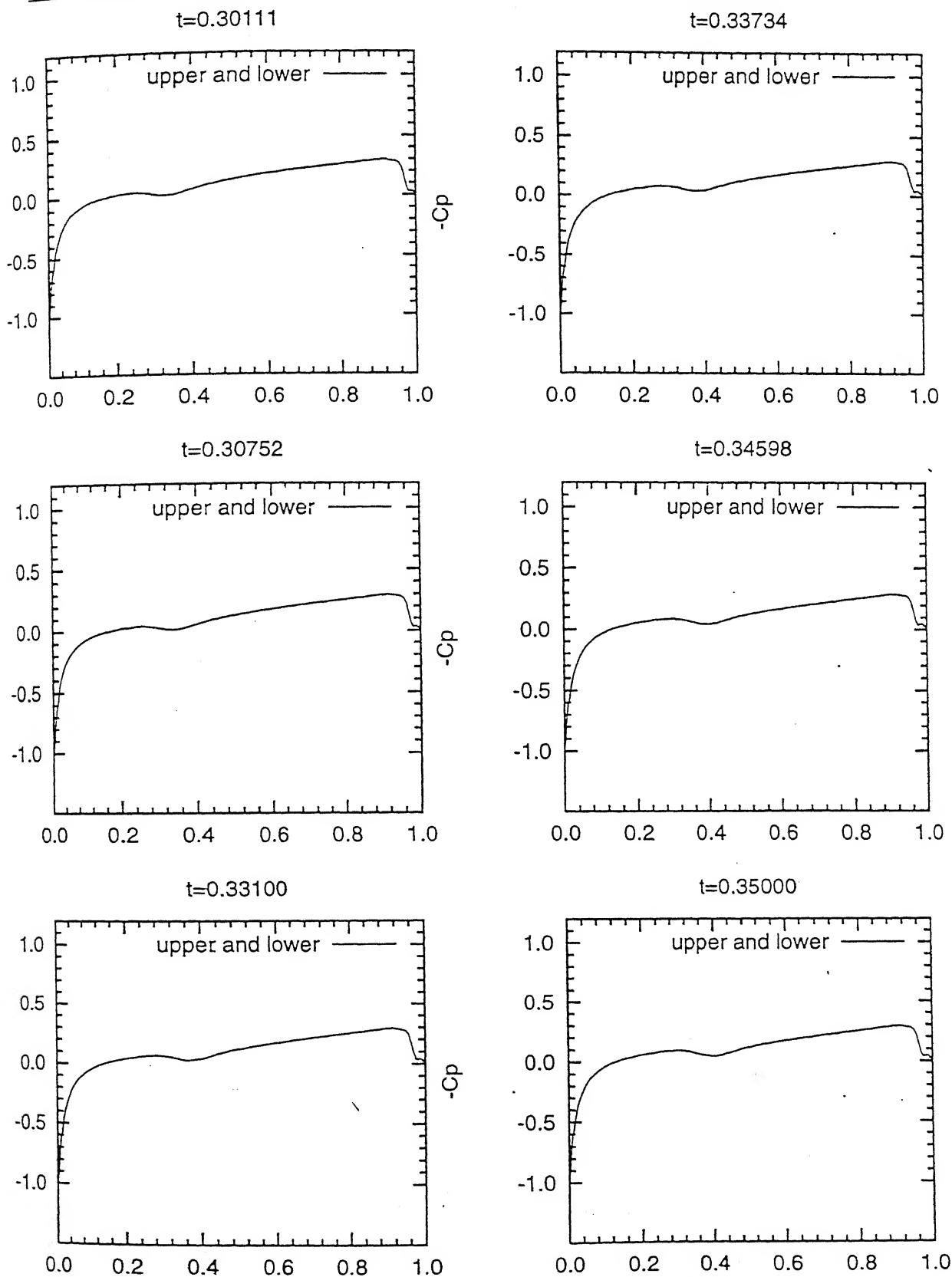


Figure 4.9: Pressure Variation for NACA0012 Navier-Stokes Solution. Mach No.= 0.8, CFL = 10^{-5}

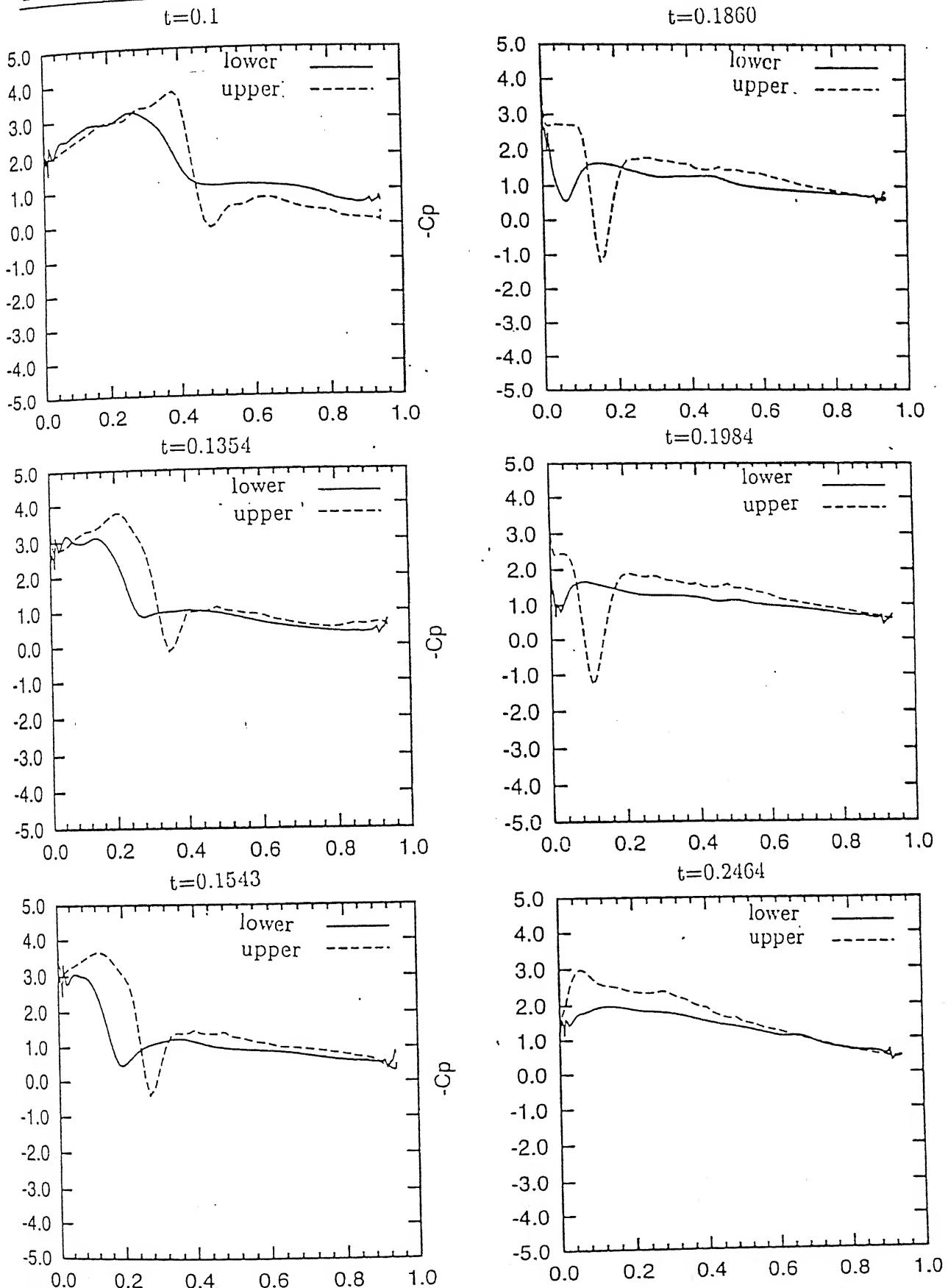
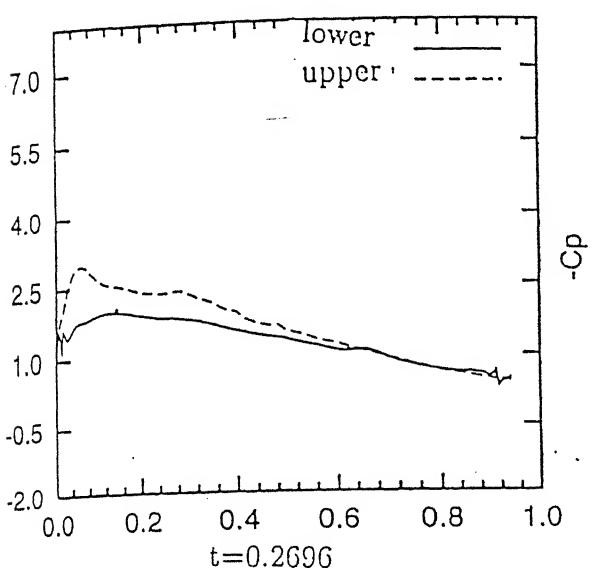
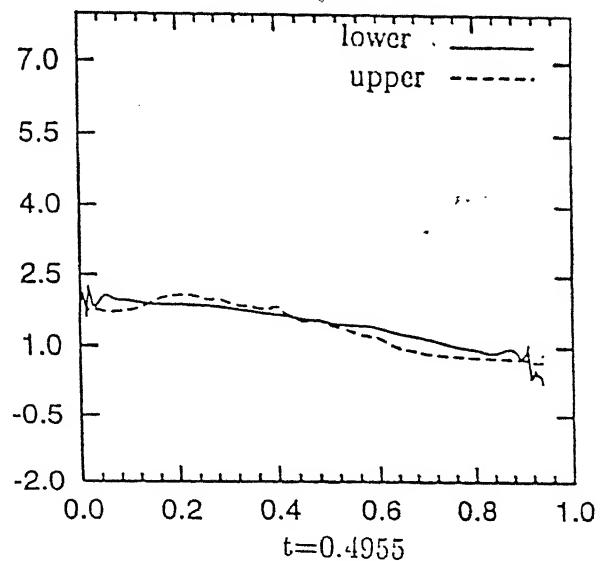
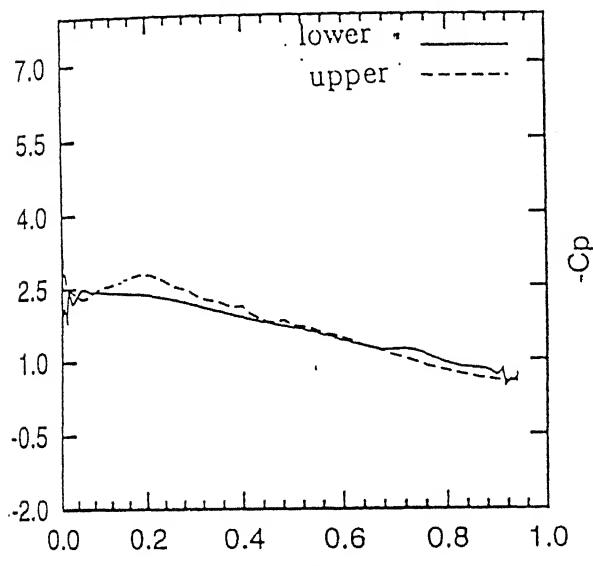
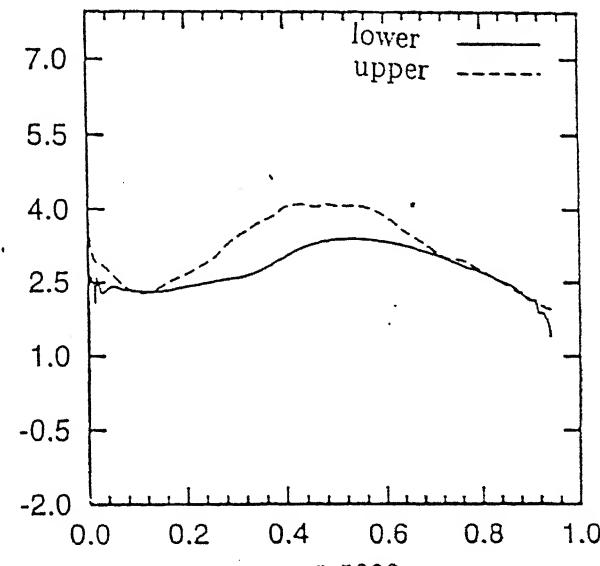
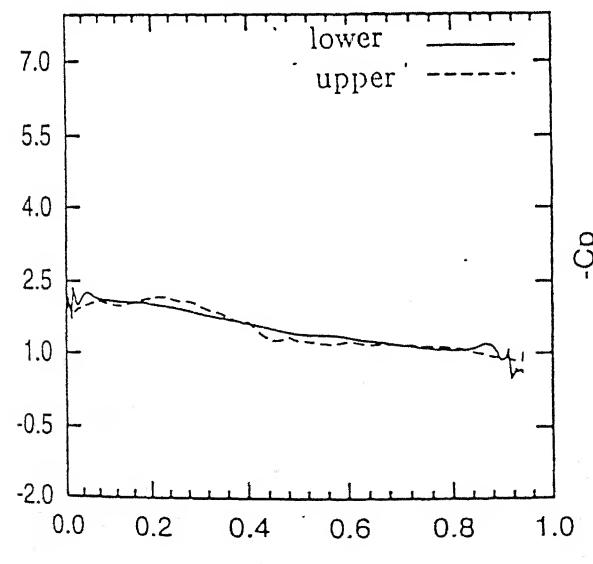
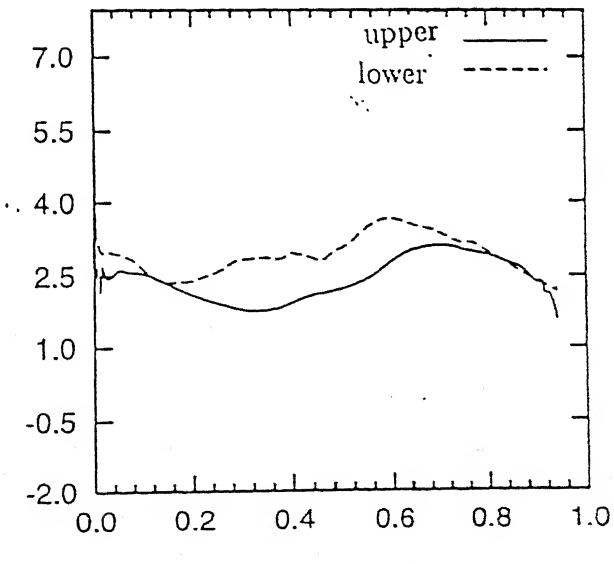
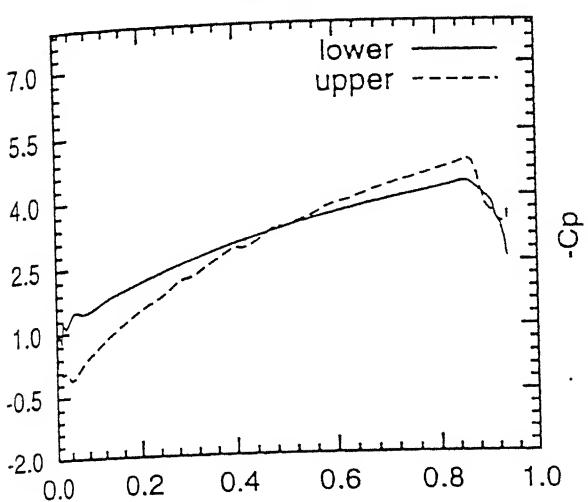


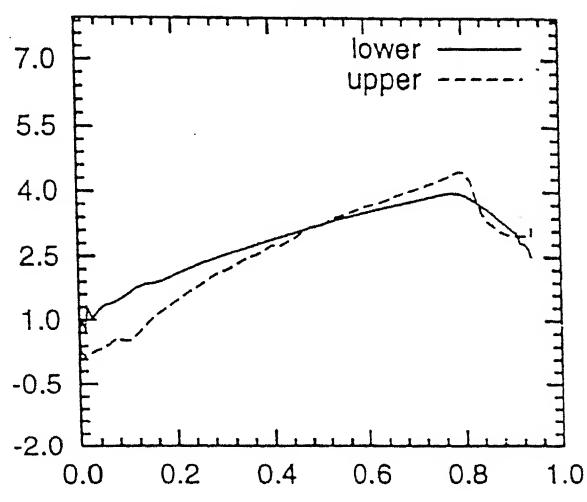
Figure 4.10: Pressure Variation for DCA Navier-Stokes Solution *Contⁿ*. Mach No.=0.3, CFL= 10^{-4}

$t=0.2464$  $t=0.3372$  $t=0.2696$  $t=0.4955$  $t=0.3220$  $t=0.5028$ Figure 4.11: Pressure Variation for DCA Navier-Stokes Solution *Contⁿ*. MachNo.=0.3, CFL= 10^{-4}

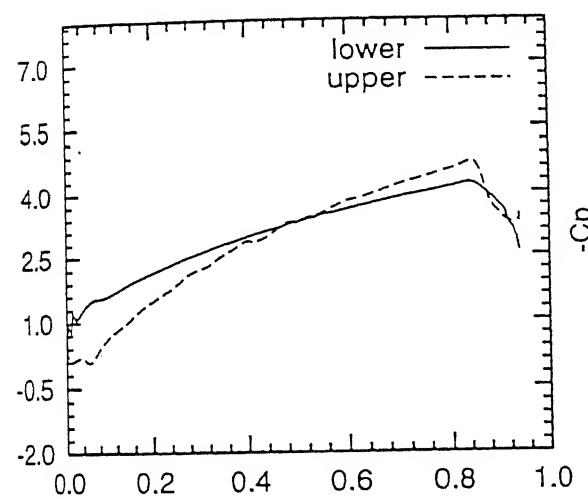
t=0.6121



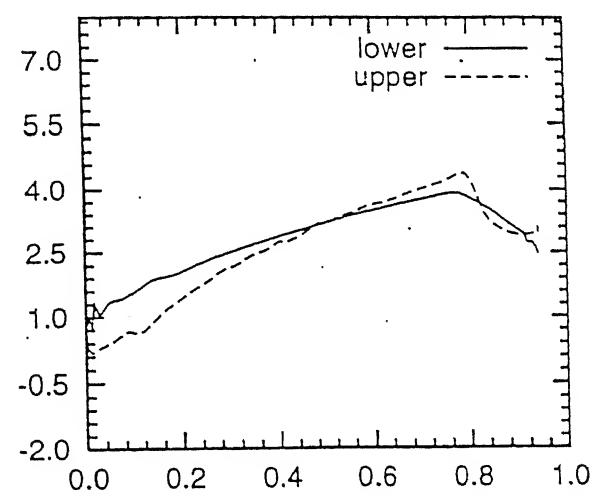
t=0.7263



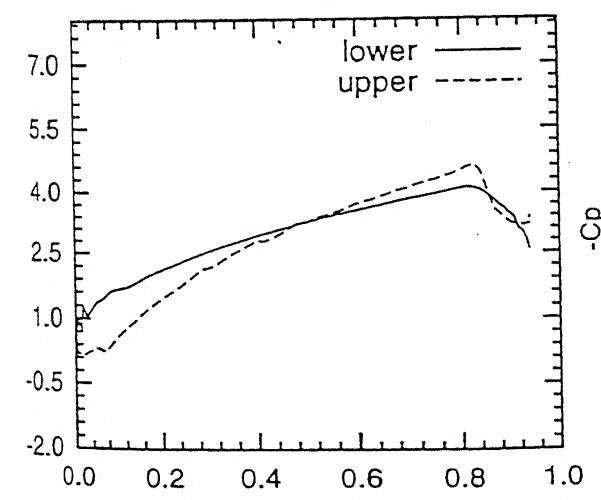
t=0.6424



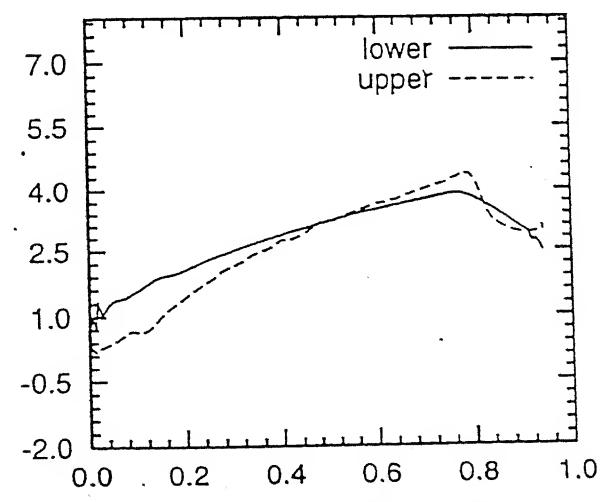
t=0.7461



t=0.6735



t=0.7498

Figure 4. Pressure Variation for DCA Navier-Stokes Solution Contⁿ. Mach

No.=0.1

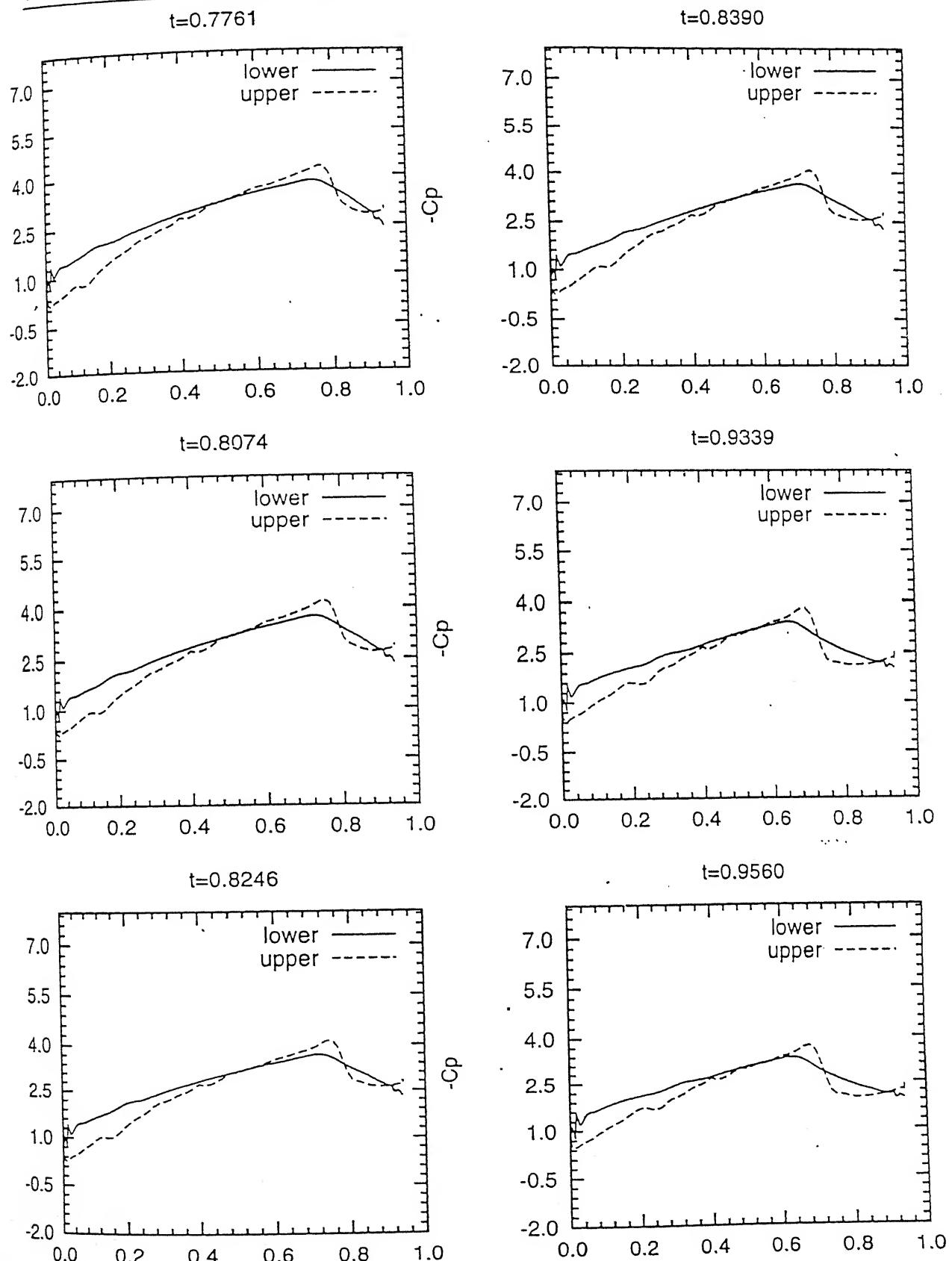
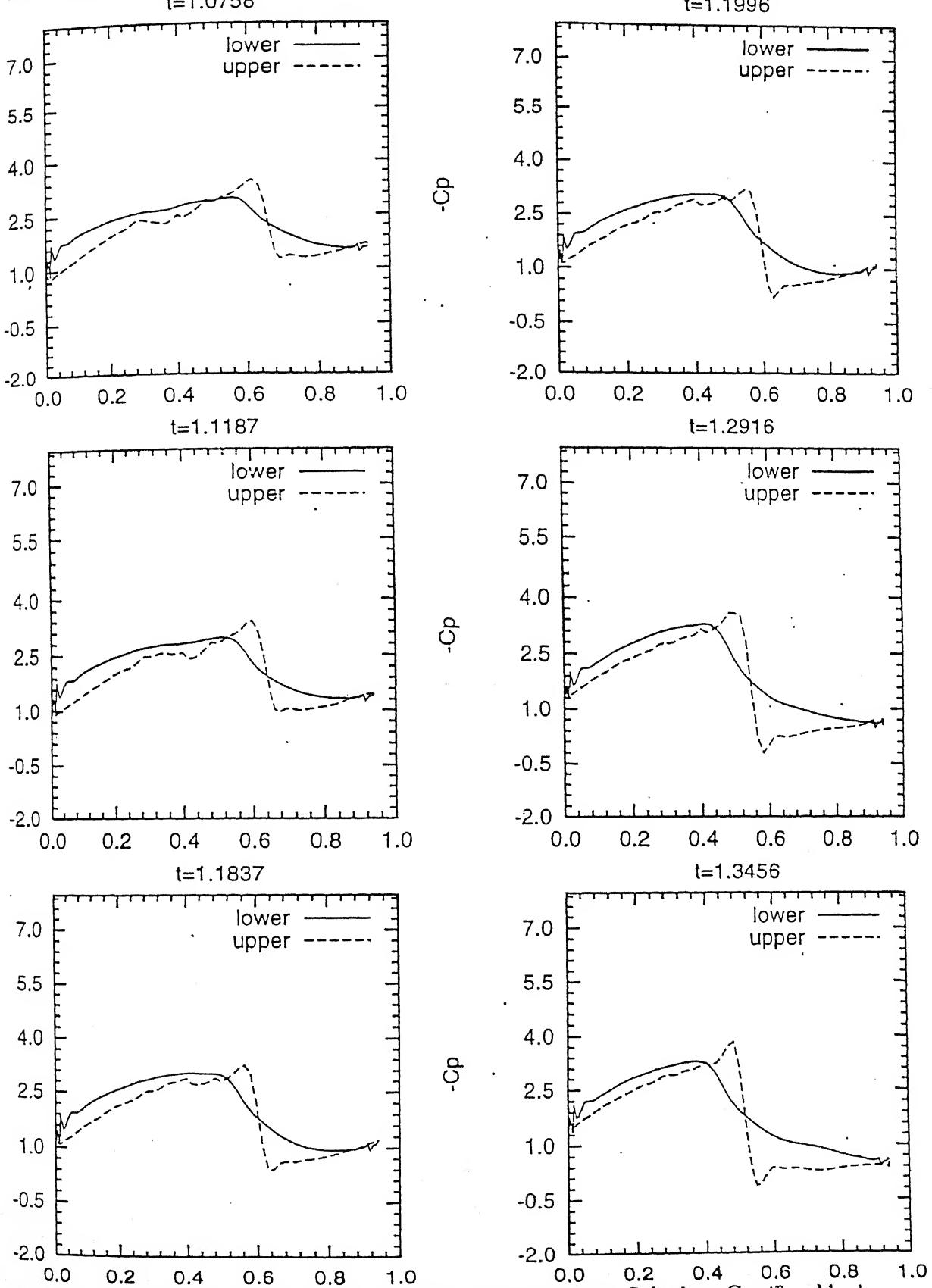
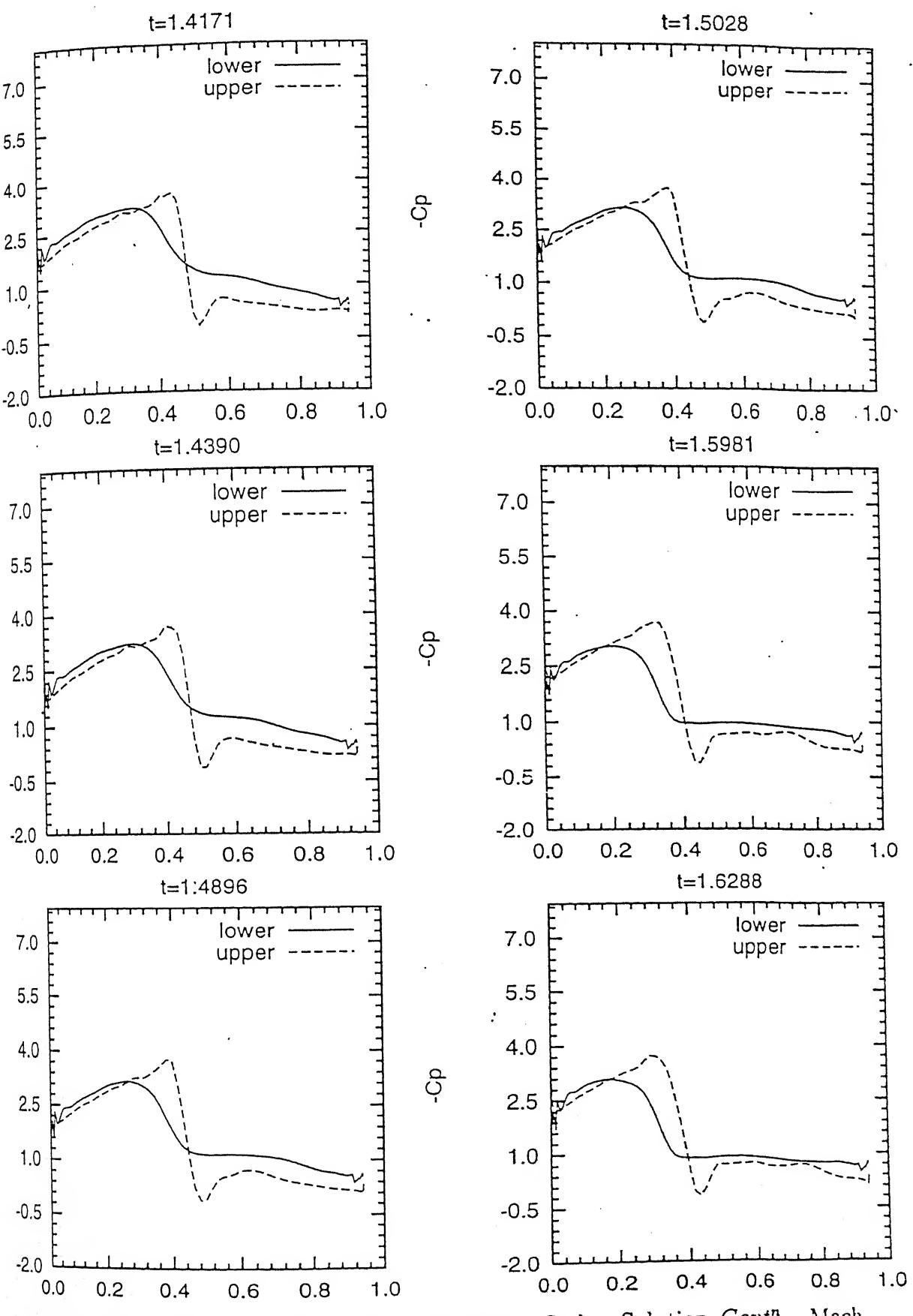


Figure 4.13: Pressure Variation for DCA Navier-Stokes Solution *Contⁿ*. Mach No.=0.3, CFL = 10^{-4}

Figure 4.14: Pressure Variation for DCA Navier-Stokes Solution *Contⁿ*.No.=0.3, CFL= 10^{-4}

Figure 4.15: Pressure Variation for DCA Navier-Stokes Solution *Cont'n.* MachNo.=0.3, CFL= 10^{-4}

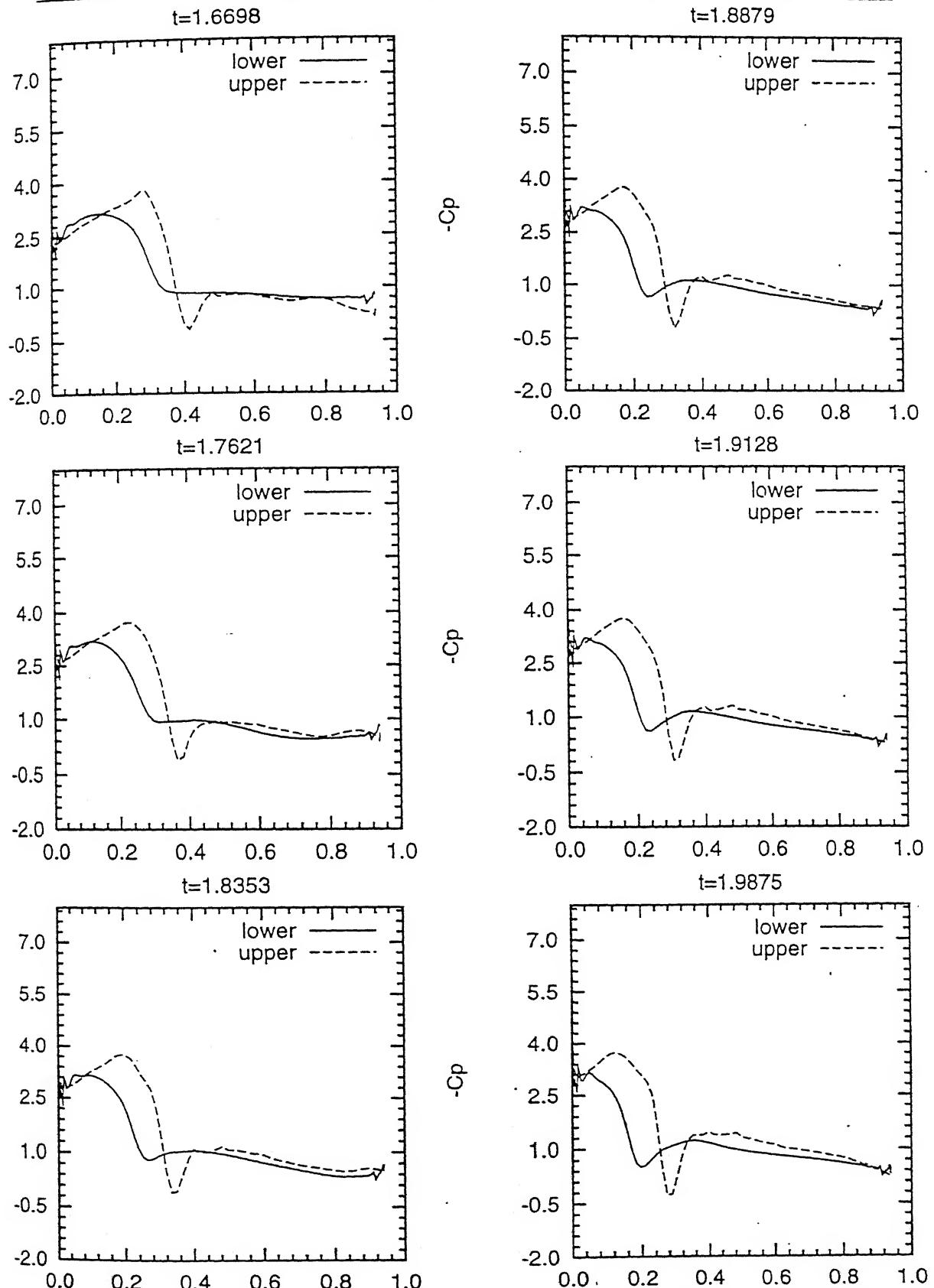


Figure 4.16: Pressure Variation for DCA Navier-Stokes Solution $Cont^n$. Mach No.=0.3, $CFL=10^{-4}$

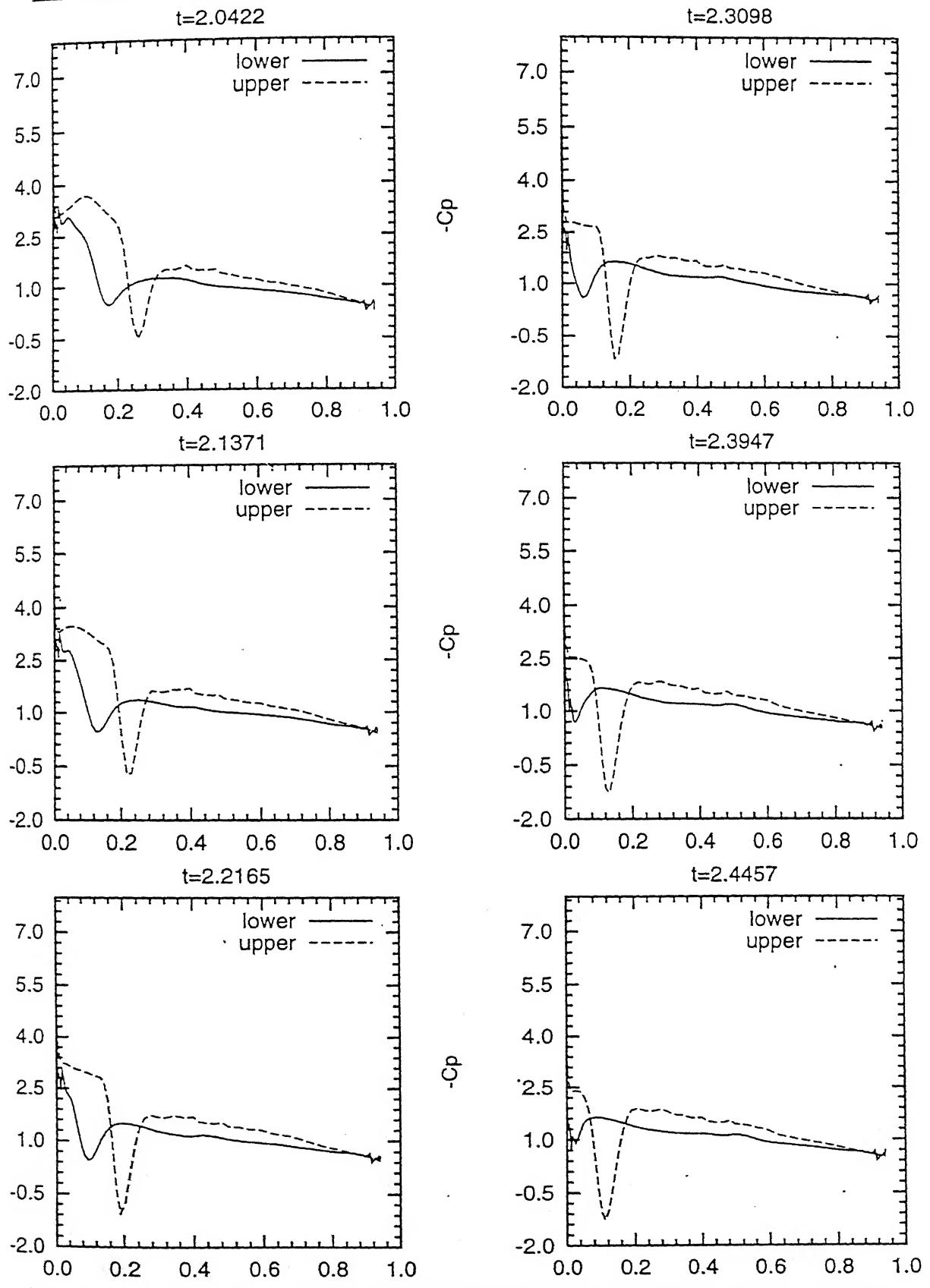


Figure 4.17: Pressure Variation for DCA Navier-Stokes Solution. Mach No.=0.3,
 $CFL = 10^{-4}$

References

- [1] FLETCHER, C.A.J., *Computational Techniques for Fluid Dynamics*, Vol. 1 and 2, Springer-Verlag, Berlin Heidelberg, New York (1987).
- [2] THOMPSON, J.F., THAMES, F.C. AND MASTIN, C.W., TOMCAT- A code for Numerical Generation of Boundary fitted curvilinear coordinate System on Field Containing any Number of Arbitrary Two-Dimensional Bodies. *Journal of Computational Physics*, **24**, 274-302 (1977).
- [3] SHU, C.W., ERIEBACHER, G., ZANG, T.A., WHITAKER, D. AND OSHER, S., High-order ENO Schemes applied to Two and Three Dimensional Compressible Flow, ICASE Report No. 91-38.
- [4] STEGER, J.L. AND WARMING, R.F., Flux vector splitting of the Inviscid Gasdynamics Equation with Application to Finite Difference Methods, *Journal of Computational Physics*, **40**, No.2, April 1981, pp. 263-293.
- [5] JAMESON, A. AND YOON, S., An LU-SSOR Scheme for the Euler and Navier-Stokes Equations, **AIAA -87-0600**.
- [6] HOHEISEL, H. AND SEYB, N.J., High Subsonic Compressor Cascade DCA edited by FOTTNER, L. in *Test Cases for Computations of Internal Flows in Aeroengine Components*, AGARD Report No. 275.

- [7] MANDAL, J.C. AND BABU, G., Transonic Cascade Flow Computations and Boundary Conditions, *48th Annual General Meeting Proceedings* (1997).
- [8] MANDAL, J.C., *Private communication regarding the NACA 0012 straight cascade transonic flow solution.*
- [9] HIRSCH, C., *Numerical Computations of Internal and External Flows*, vol.1 and 2, John Wiley and Sons (1987).
- [10] MORINISHI, K. AND SATOFUKA, N., Convergence Acceleration of the Rational Runge-kutta Scheme for the Euler and Navier-Stokes Equations, *Computers and Fluids*, **19**, No.3/4, pp. 305-313 (1991).
- [11] RAVICHANDRAN, K.S., Higher Order KFVS Algorithms Using Compact Upwind Difference Operators, *Journal of Computational Physics*, **130**, 161-173 (1997).
- [12] PULLIAM, T.H. AND STEGER, J.L., Implicit Finite Difference Simulations of Three Dimensional Compressible Flow, *AIAA*, **18**, No.2, pp. 159-167 (1980).
- [13] KALLINDERIS, J.G. AND BARON, J.R., *Computational Methods in Viscous Aerodynamics*, ch.4, Elsevier and Computational Mechanics Publications, Boston (1990).
- [14] SENGUPTA, T.K. AND SENGUPTA, R., Flow Past an Impulsively Started Circular Cylinder at High Reynolds Number, *Computational Mechanics*, **14**, No.4, pp.298-310 (1994).

- [15] ANDERSON, J.D., *Modern Compressible Flow*, McGraw-Hill Publishing Company, Singapore (1990).
- [16] COURANT R. AND FRIEDRICH K.O., *Supersonic Flow and shock Waves*, Interscience Publication, Inc. (1967).
- [17] WHITE, F.M., *Viscous Fluid Flow*, McGRAW-HILL, Inc. (1991).
- [18] VERDON, J.M., Unsteady Aerodynamics for Turbomachinery Aeroelastic Applications edited by NIXON, D. in *Unsteady Transonic Aerodynamics*, AIAA, 120, Washington (1988).
- [19] DECONINCK, H., *Upwind Methods and Multidimensional Splittings for the Euler Equations*, Von Karman Institute for Fluid Dynamics Lecture Series 1991 – 01.
- [20] WATERSON, N.P. AND DECONINCK H., A unified approach to the design and application of bounded higher-order convective schemes in *Numerical Methods in Laminar and Turbulent Flow '95*, Vol.IX, Part 1, Section 1/18, Pineridge Press (1995).